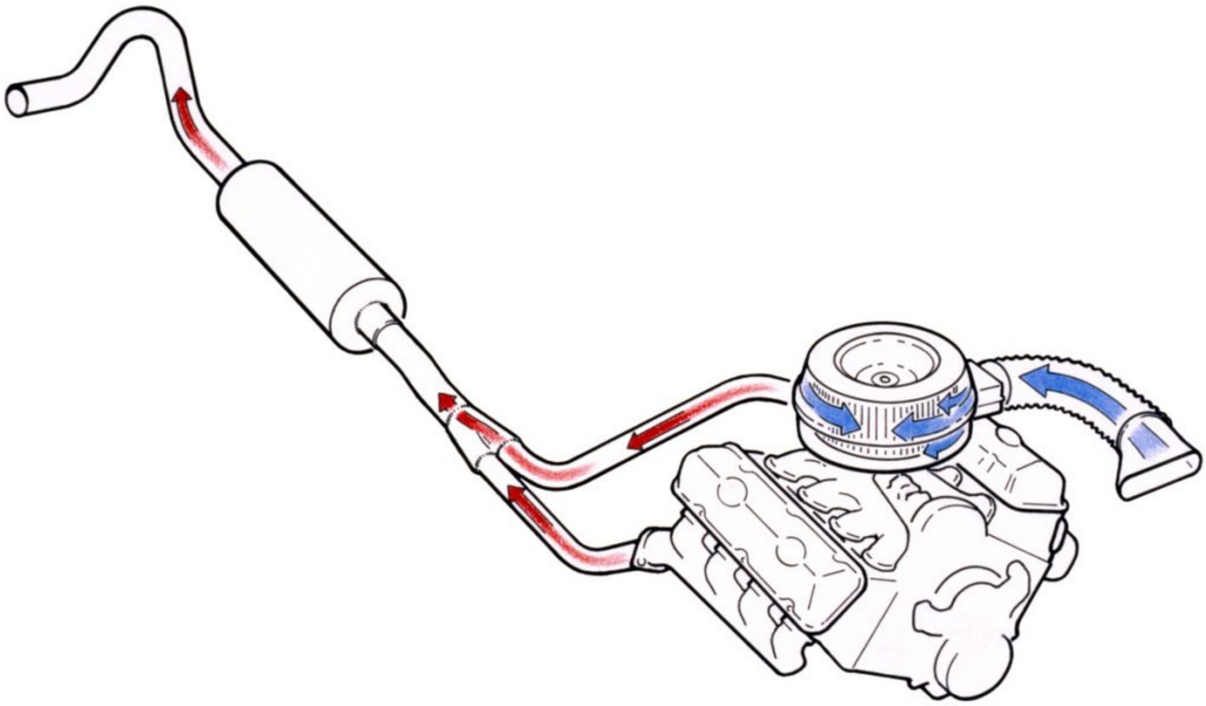
EE5103 Linear System Mini Project

Controller Design for Diesel Engine Air System Control



A0260014Y
Guo Shiping

Contents

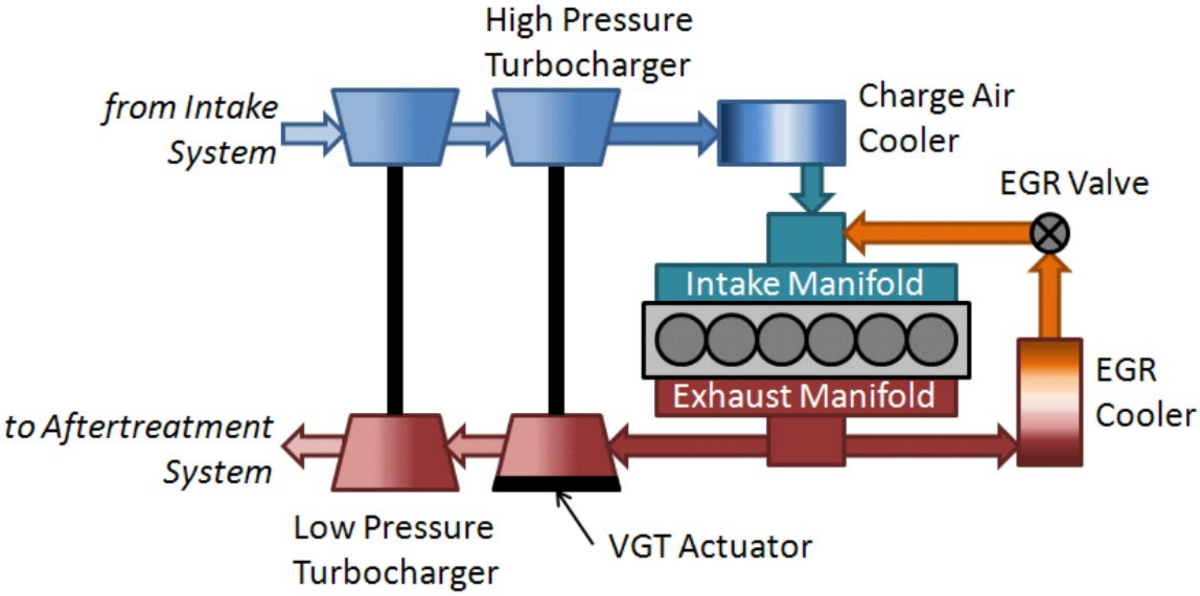
Abstract

This mini-project explores advanced control strategies for emissions reduction in diesel engines, focusing on active control of air and exhaust gas. Strategies include Pole Placement Feedback, Linear Quadratic Regulator (LQR), and observer-based LQR, with simulations demonstrating their effectiveness. Challenges in achieving stability with decoupling controllers are discussed, and alternative output feedback methods are considered. A servo controller is designed for precise system control around a set point, showcasing effectiveness in tracking and disturbance mitigation. The findings contribute insights into optimal performance and compliance with emissions standards.

Keywords：Pole placement, LQR, State observer, Decoupling control, Servo control

1. Introduction

As diesel engine emissions standards become increasingly stringent, one of the most commonly employed method of emissions reduction by engine manufacturers is active control of inducted air and recirculated exhaust gas (EGR). Commonly, actuators like EGR (Exhaust Gas Recirculation) and VGTs (Variable Geometry Turbochargers) are employed in order to manipulate the flow of gases through a diesel engine to achieve the desired reduction in NOx and PM emissions, as mandated by the US government other governments around the world. A diagram of a common air path layout involving this kind of control strategy is illustrated in Figure 1, where one can see the interactions and internal feedback caused by the EGR loop and turbochargers.



In practice, the movement of either one of these actuators has a direct impact on the percentage of EGR in the intake manifold as well as the amount of air that can be inducted into the cylinder, resulting in a complex, interactive, multivariable system. Another difficulty of the engine air path control is to balance the engine's customer performance against the mandates of government agencies to reduce emissions, since the introduction of EGR into the air system has a negative impact on the overall fuel economy of the engine. In this mini project, we will model this engine air system as a MIMO linear system and try to control it using the techniques learned in *Linear Systems*

2. State space Model

Given state space equations:

$$\dot{x} = Ax + Bu$$
$$y = Cx,$$

$$A = \begin{bmatrix} -8.8487 + \frac{a-b}{5}, & -0.0399, & -5.5500 + \frac{c+d}{10}, & 3.5846 \\ -4.5740, & 2.5010 * \frac{d+5}{c+5}, & -4.3662, & -1.1183 - \frac{a-c}{20} \\ 3.7698, & 16.1212 - \frac{c}{5}, & -18.2103 + \frac{a+d}{b+4}, & 4.4936 \\ -8.5645 - \frac{a-b}{c+d+2}, & 8.3742, & -4.4331, & -7.7181 * \frac{c+5}{b+5} \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0564 + \frac{b}{10+c}, & 0.0319 \\ 0.0165 - \frac{c+d-5}{1000+20a}, & -0.02 \\ 4.4939, & 1.5985 * \frac{a+10}{b+12} \\ -1.4269, & -0.2730 \end{bmatrix}$$

$$C = \begin{bmatrix} -3.2988, & -2.1932 + \frac{10c+d}{100+5a}, & 0.0370, & -0.0109 \\ 0.2922 - \frac{ab}{500}, & -2.1506, & -0.0104, & 0.0163 \end{bmatrix}$$

My student number is **A0260014Y**.
So the final four number are 0 0 1 4, a = 0, b= 0, c =1, d = 4

$$Initial\ condition: \begin{bmatrix} 0.5 & -0.1 & 0.3 & -0.8 \end{bmatrix}^T$$

$$A = \begin{bmatrix} -8.8487 & -0.0399 & -5.0500 & 3.5846 \\ -4.5740 & 3.7515 & -4.3662 & -1.0683 \\ 3.7698 & 15.9212 & -17.2103 & 4.4936 \\ -8.5645 & 8.3742 & -4.4331 & -9.2617 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0564 & 0.0319 \\ 0.0165 & -0.0200 \\ 4.4939 & 1.3321 \\ -1.4269 & -0.2730 \end{bmatrix}$$

$$C = \begin{bmatrix} -3.2988 & -2.0532 & 0.0370 & -0.0109 \\ 0.2922 & -2.1506 & -0.0104 & 0.0163 \end{bmatrix}$$

3. Design specifications

The transient step response performance specifications for all the outputs *y* in state space model (1) are as follows:

1) The overshoot is less than 10%.
2) The 2% settling time is less than 20 seconds.

Note:

(a) This transient response is checked by giving a step reference signal for each input channel, i.e., [1, 0] and [0, 1], with zero initial conditions;

(b) For all the following task 1) to 5), your control system should satisfy this performance specification and you are supposed to finish the required investigation for each task as well.

Using second-order system we know that:

$$\frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$$

Rise time, $t_r = \dfrac{1.8}{\omega_n}$;

Peak overshoot, $M_p = e^{\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}}$;

2% settling time, $t_s = \dfrac{4}{\zeta\omega_n}$.

$$-0.5912 < \zeta < 0.5912$$
$$w_n >= 0.4$$

Let

$$\zeta = 0.5$$
$$w_n = 0.4$$

Solve the equation I got two solutions:

$$pole\ 1 : -0.2000 - 0.3464i$$
$$pole\ 2 : -0.2000 + 0.3464i$$

So, the dominated poles are these two poles, this system are four order, I choose the other two poles far aways from these two poles 5 times away.

4. Pole place feedback controller.

Assume that you can measure all the four state variables, design a state feedback controller using the pole place method, simulate the designed system, check the **step responses and show all the four state responses to non-zero initial state with zero external inputs**. Discuss **effects of the positions of the poles on system performance**, and also **monitor control signal size**. In this step, both the disturbance and set point can be assumed to be zero. (15 points)

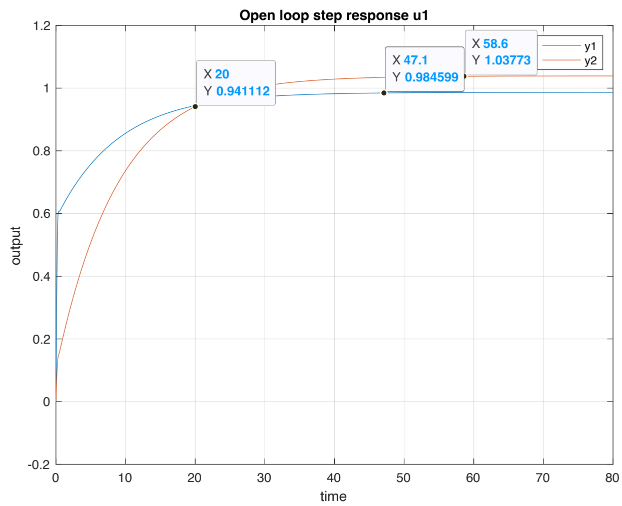


Figure 1 Open loop response u1

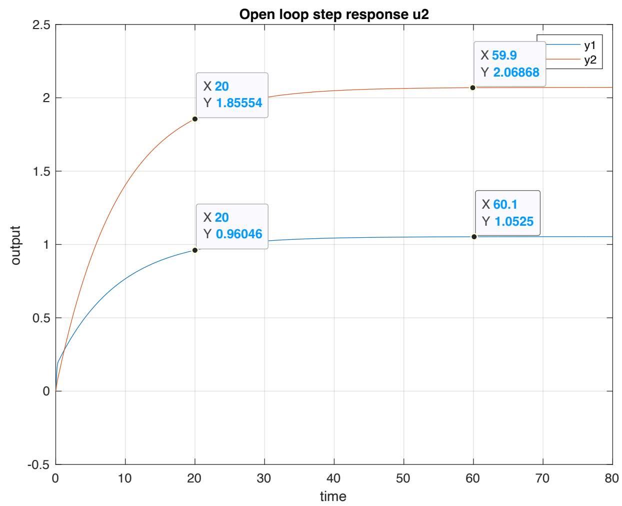


Figure 2 Open loop response u2

They don't satisfy the requirements.

4.1 Full Rank Method

4.1.1 Derivation procedure

Follow the systemic procedure:

Step1: Check the controllability:

$$W_c = \begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix}$$

Rank of controllability matrix is 4

$$b_1 \quad b_2 \quad Ab_1 \quad Ab_2 \quad A^2b_1 \quad A^2b_2 \quad A^3b_1 \quad A^3b_2$$

1	2	3	4	5	6	7	8
0.0564	0.0319	-28.3088	-7.9871	646.5038	180.1903	-8.9287e+03	-2.4287e+03
0.0165	-0.0200	-18.2930	-5.7454	431.9993	125.3762	-6.2779e+03	-1.7526e+03
4.4939	1.3321	-83.2780	-24.3505	1.0036e+03	280.3407	-5.6034e+03	-1.4142e+03
-1.4269	-0.2730	-7.0512	-3.8175	523.7475	163.5967	-1.1219e+04	-3.2513e+03

Step2: To form C

$$b_1 \quad Ab_1 \quad A^2b_1 \quad A^3b_1 \quad b_2 \quad Ab_2 \quad A^2b_2 \quad A^3b_2$$

1	2	3	4	5	6	7	8
0.0564	-28.3088	646.5038	-8.9287e+03	0.0319	-7.9871	180.1903	-2.4287e+03
0.0165	-18.2930	431.9993	-6.2779e+03	-0.0200	-5.7454	125.3762	-1.7526e+03
4.4939	-83.2780	1.0036e+03	-5.6034e+03	1.3321	-24.3505	280.3407	-1.4142e+03
-1.4269	-7.0512	523.7475	-1.1219e+04	-0.2730	-3.8175	163.5967	-3.2513e+03

This is a MIMO system, need to select 4 independent vectors out of 4 times 2 vectors from the controllability matrix in the strict order from left to right and group them in a square matrix C in the following form.

The first four columns are independent.

1	2	3	4
0.0564	-28.3088	646.5038	-8.9287e+03
0.0165	-18.2930	431.9993	-6.2779e+03
4.4939	-83.2780	1.0036e+03	-5.6034e+03
-1.4269	-7.0512	523.7475	-1.1219e+04

C Inverse:

1	2	3	4
8.0718	-27.4396	2.6390	7.6124
2.2036	-5.7838	0.4008	1.2825
0.1802	-0.4291	0.0259	0.0838
0.0060	-0.0129	6.2204e-04	0.0020

The controllable canonical form can then be computed from this matrix C

Step3: To get T

This state space form is not controllable canonical form need to use a matrix T to transform the system into canonical form. There are multiple inputs. If want all the inputs to be applied to the system efficiently, then more rows will be taken out in addition to the last row. **For each input, there will be one row to be taken out to construct T.**

4 vectors in C are associated with u1.

0 vectors in C are associated with u2.

$$T = \begin{bmatrix} q_1T \\ q_1TA \\ q_1TA^2 \\ q_1TA^3 \end{bmatrix}$$

1	2	3	4
8.0718	-27.4396	2.6390	7.6124
-1.1633	2.5021	-0.1206	-0.3968
1.7930	4.1909	-1.2159	-3.7101
-7.8431	-34.7778	10.0211	30.8483

Obtained C and form T

$$\bar{A} = TAT^{-1}\bar{B} = TB$$

1	2	3	4
3.1308e-14	1.0000	5.0071e-14	2.0227e-15
1.6716e-12	1.5120e-11	1.0000	1.2490e-13
-1.0850e-11	-1.0164e-10	-2.2453e-11	1.0000
-193.8328	-1.7643e+03	-409.0714	-31.5692

$$\overline{B} = TB$$

1	2	3	4
3.1308e-14	1.0000	5.0071e-14	2.0227e-15
1.6716e-12	1.5120e-11	1.0000	1.2490e-13
-1.0850e-11	-1.0164e-10	-2.2453e-11	1.0000
-193.8328	-1.7643e+03	-409.0714	-31.5692

There are two inputs, so there are two nontrivial rows since the inputs only affect the nontrivial rows!

Design the feedback gain matrix for the controllable canonical form

$$\bar{K} = \begin{bmatrix} \bar{k}_{11} & \bar{k}_{12} & \bar{k}_{13} & \bar{k}_{14} \\ \bar{k}_{21} & \bar{k}_{22} & \bar{k}_{23} & \bar{k}_{24} \end{bmatrix}$$

Form the closed loop matrix:

Desired eigenvalues

$$(s+2)(s+2)(s+0.2+0.3465i)(s+0.2-0.3465i)$$

$$s^4 + s^3\left(\frac{22}{5} + \frac{1i}{10000}\right) + s^2\left(\frac{14400069}{2500000} + \frac{21i}{50000}\right) + s\left(\frac{1400069}{625000} + \frac{3i}{6250}\right) + \left(\frac{400069}{625000} + \frac{1i}{12500}\right)$$

Specify a desired closed-loop matrix

$$\bar{A} = \bar{B}\bar{K}$$

The new feed back system is

```
ans =

[      3.13e-14 - 2.24*k21 - 1.0*k11,          1.0 - 2.24*k22 - 1.0*k12,       5.01e-14 - 2.24*k23 - 1.0*k13,       2.02e-15 - 2.24*k24 - 1.0*k14]
[4.03e-14*k11 + 0.139*k21 + 1.67e-12, 4.03e-14*k12 + 0.139*k22 + 1.51e-11,      4.03e-14*k13 + 0.139*k23 + 1.0, 4.03e-14*k14 + 0.139*k24 + 1.25e-13]
[3.12e-12*k11 + 0.633*k21 - 1.08e-11, 3.12e-12*k12 + 0.633*k22 - 1.02e-10, 3.12e-12*k13 + 0.633*k23 - 2.25e-11,      3.12e-12*k14 + 0.633*k24 + 1.0]
[     3.3e-11*k11 - 5.37*k21 - 194.0,      3.3e-11*k12 - 5.37*k22 - 1760.0,      3.3e-11*k13 - 5.37*k23 - 409.0,       3.3e-11*k14 - 5.37*k24 - 31.6]
```

Compare the non-trivial rows of the canonical form with feedback to the desired one

Compute the original feedback gain K

$$K = \bar{K}T$$

4.1.2 Simulation results

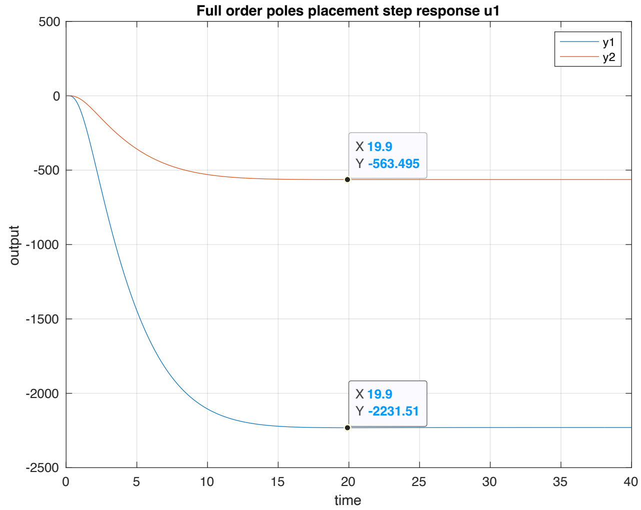


Figure 3 Full rank method step response with input u1

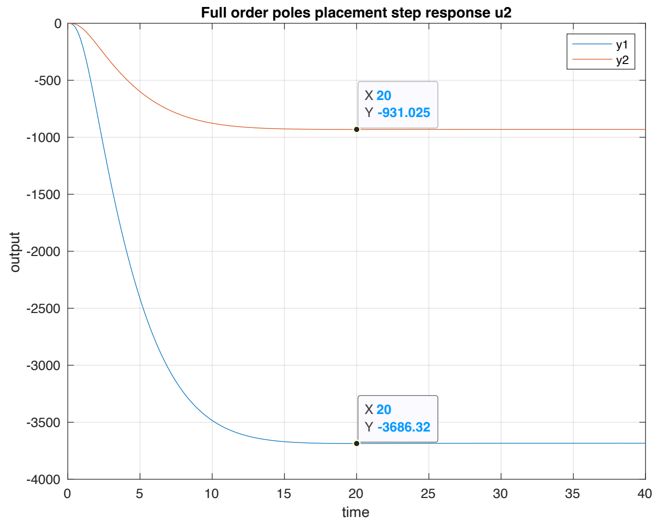


Figure 4 Full rank method step response with input u2

4.2 Unity Rank Method

4.2.1 Derivation procedure

Multiple input system can be transferred to single input form and solve using SISO method. In this method we need choose q first, which is

$$u = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} v = qv$$

v is scalar.

$$\dot{x} = Ax + Bu = Ax + Bqv$$

Let

$$q = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$v = -k^T x$$

$$u = qv = -qk^T x = -Kx$$

Let

$$k = \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix}$$

$$K = qk^T = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \begin{bmatrix} k_1 & k_2 & k_3 & k_4 \end{bmatrix}$$

The desired poles are the same as full rank method:

$$A - BK = A - Bqk = \begin{bmatrix} -8.7170 & -0.5610 & -4.5728 & 3.6749 \\ -4.5998 & 3.8534 & -4.4595 & -1.0860 \\ 11.6141 & -15.1117 & 11.2075 & 9.8713 \\ -10.7266 & 16.9275 & -12.2656 & -10.7439 \end{bmatrix}$$

4.2.2 Simulation results

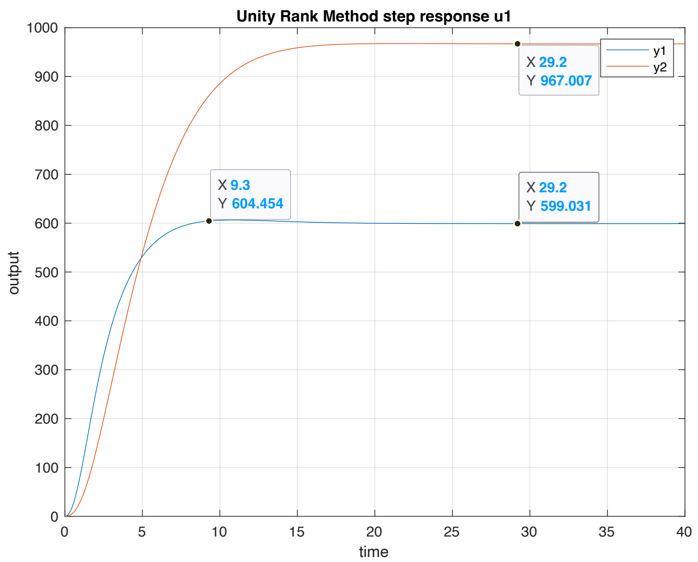


Figure 5 Unity rank method step response with input u1

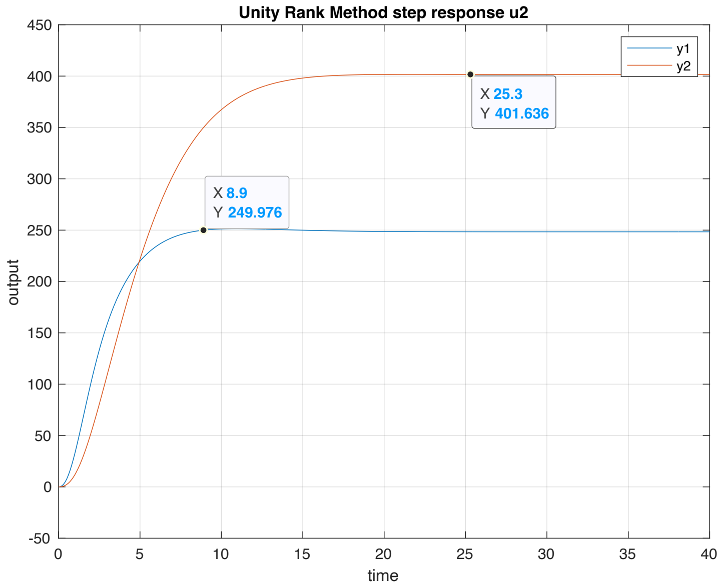


Figure 6 Unity rank method step response with input u2

Close loop step response:

Show all the four state responses to non-zero initial state with zero external inputs.

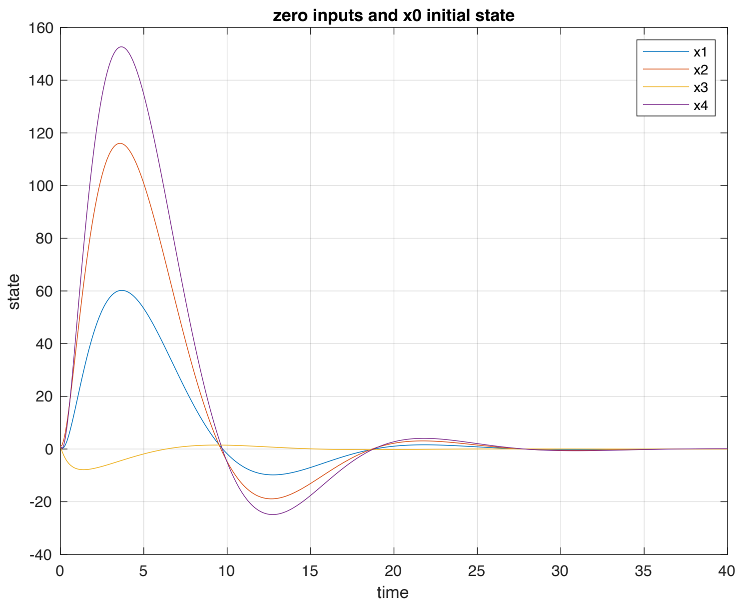


Figure 7 State responses to non-zero initial state

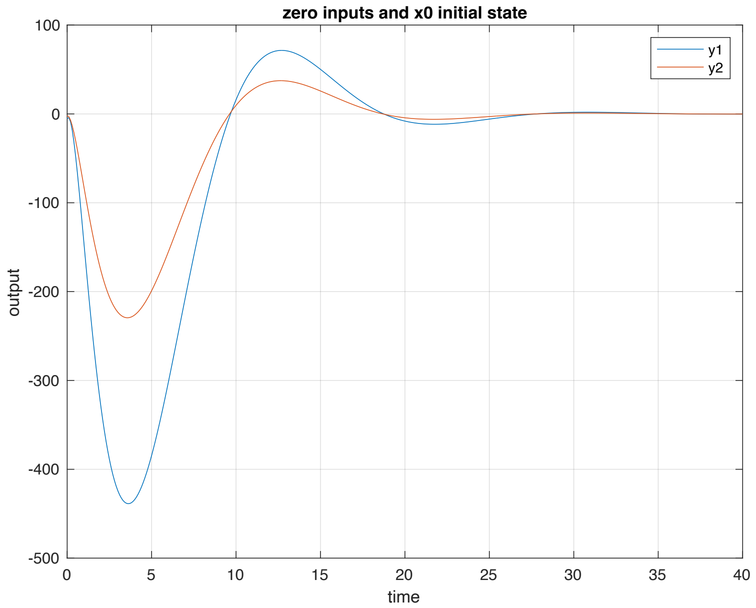


Figure 8 Output to non-zero initial state

Use Simulink to double check the results:

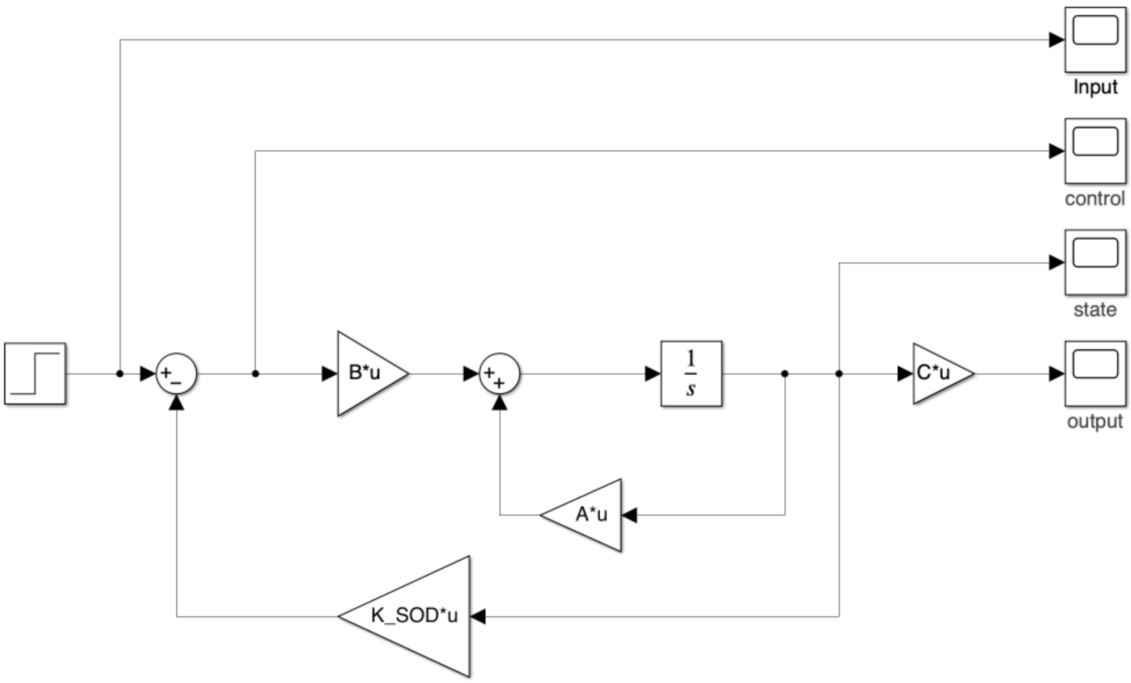


Figure 9 Full rank method Simulink simulation structure

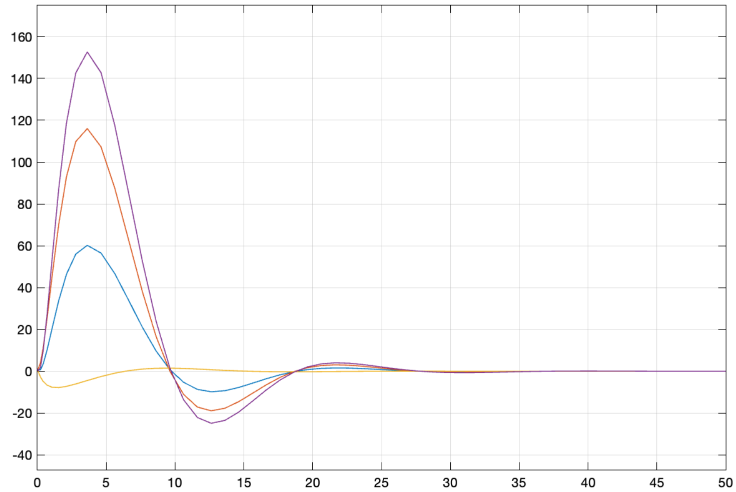


Figure 10 State responses to non-zero initial state in Simulink

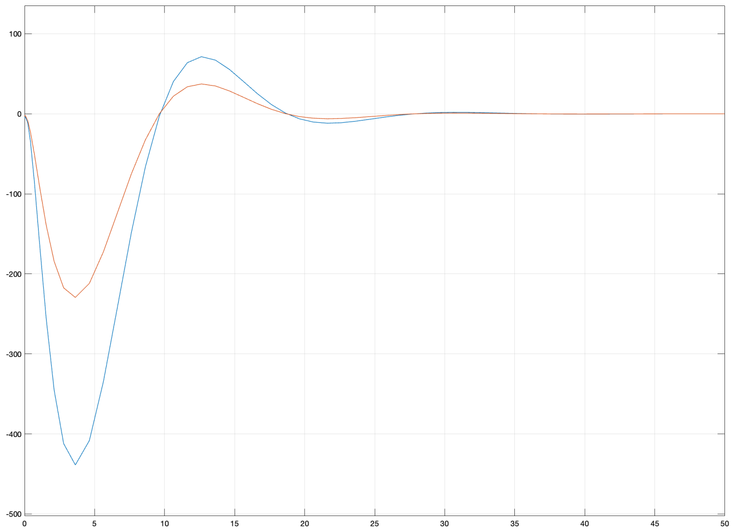


Figure 11 Output to non-zero initial state in Simulink

Discuss effects of the positions of the poles on system performance

In this context, I've introduced artificial poles at various locations and conducted a comparative analysis of the outcomes.

Step response:

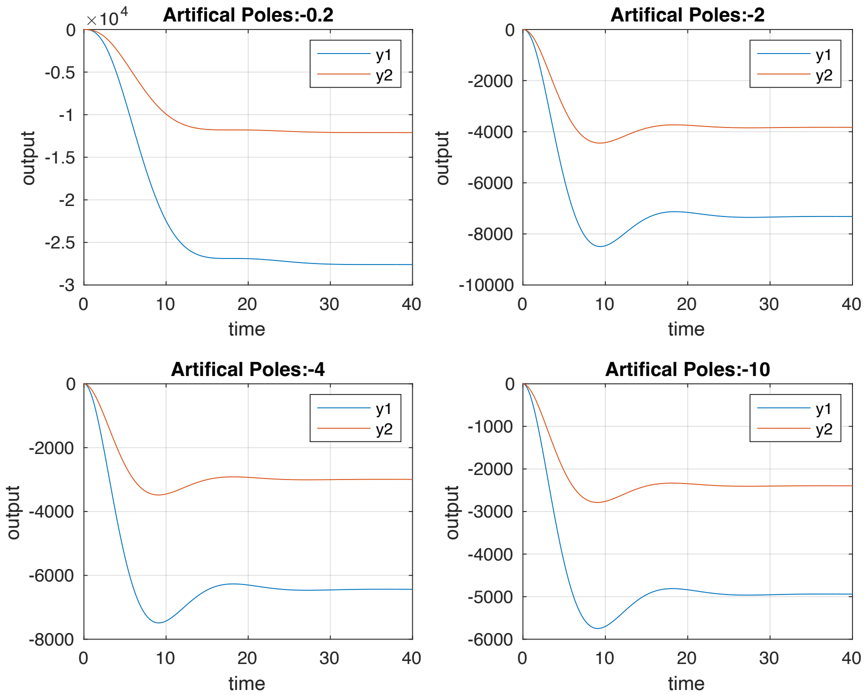


Figure 12 The step response under different places of poles

No input

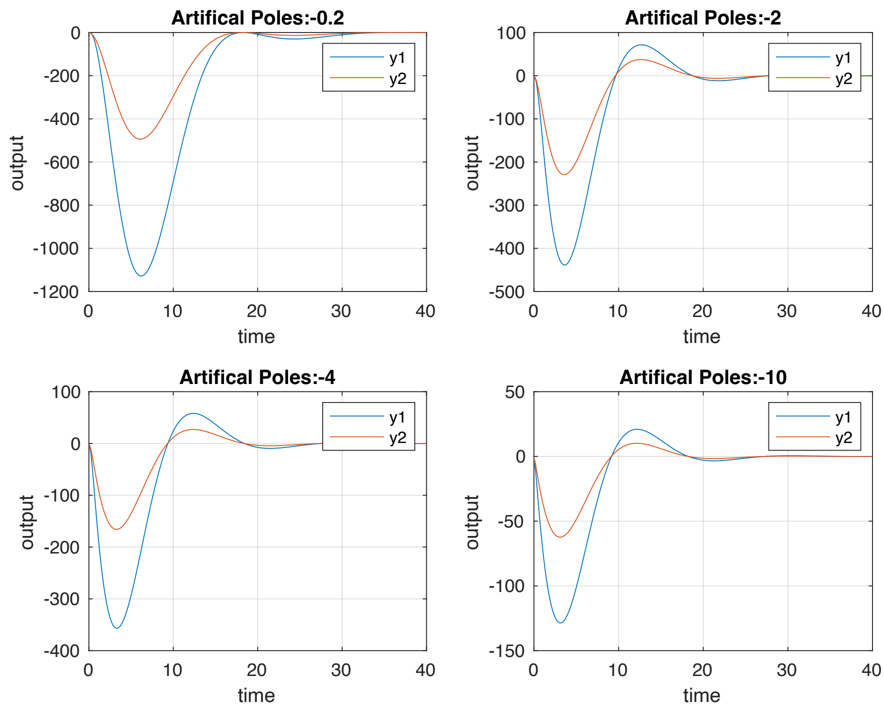


Figure 13 Response with non-zero state under different places of poles

States:

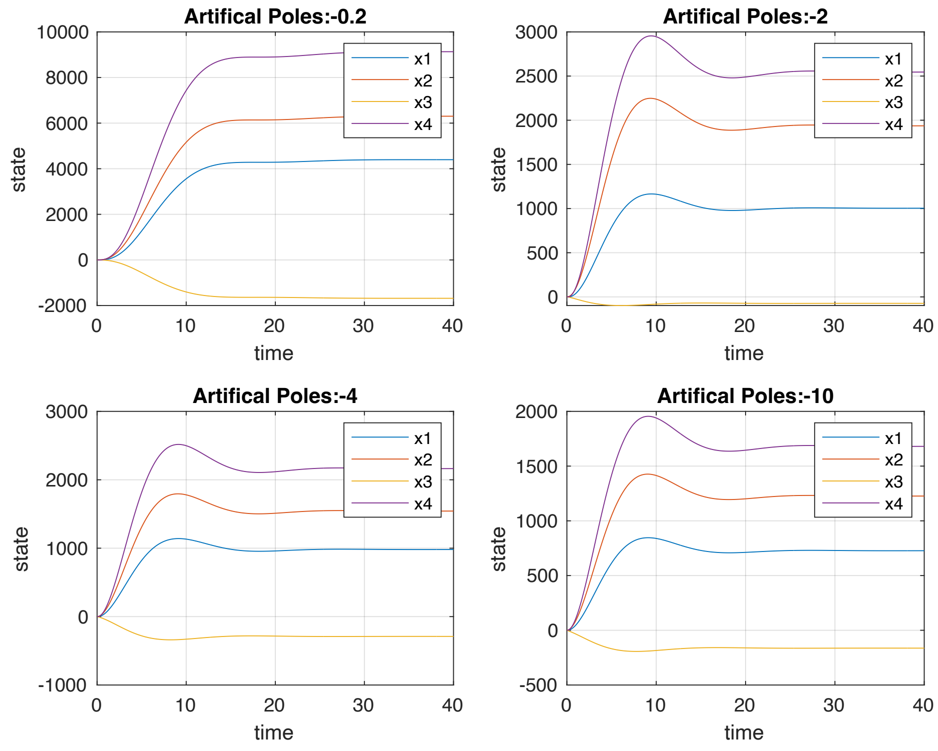


Figure 14 State response under different places of poles

No input:

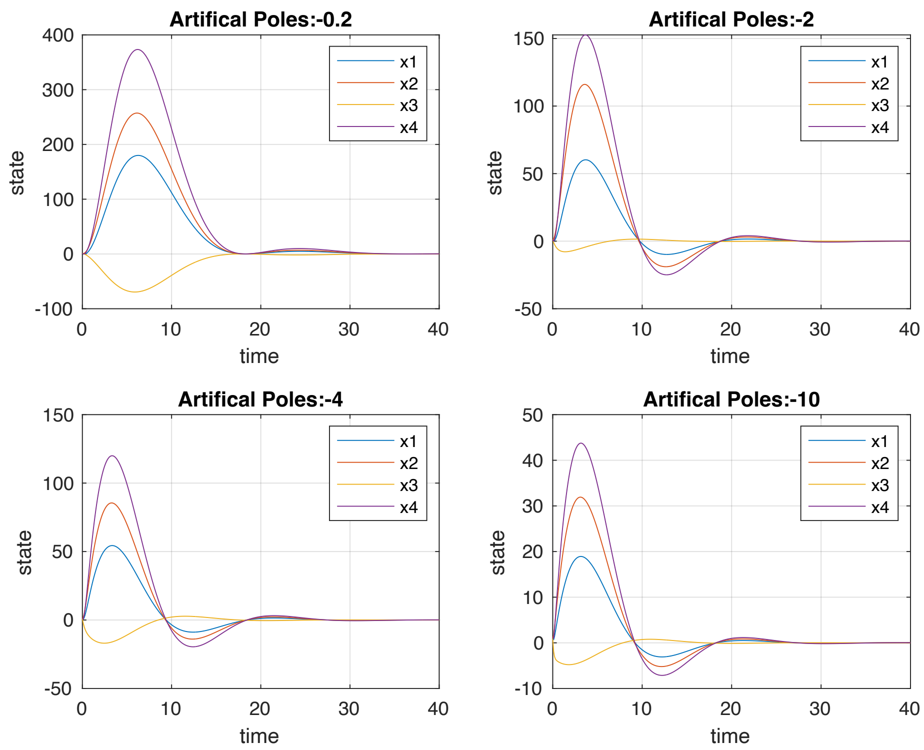


Figure 15 State response with non-zero state under different places of poles

Monitor control signal size

Step response control signal size:

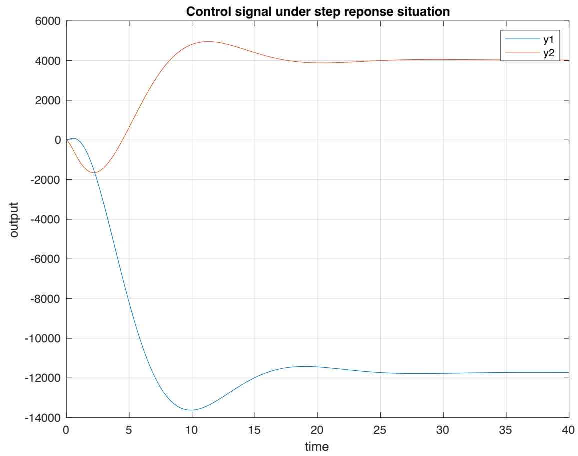


Figure 16 Control signal under step response situation

Zero input with x0 initial state control signal:

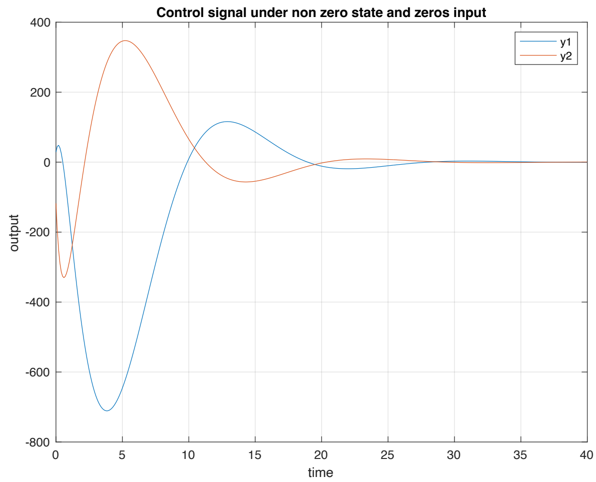


Figure 17 Control signal with non-zero initial state

Control signal

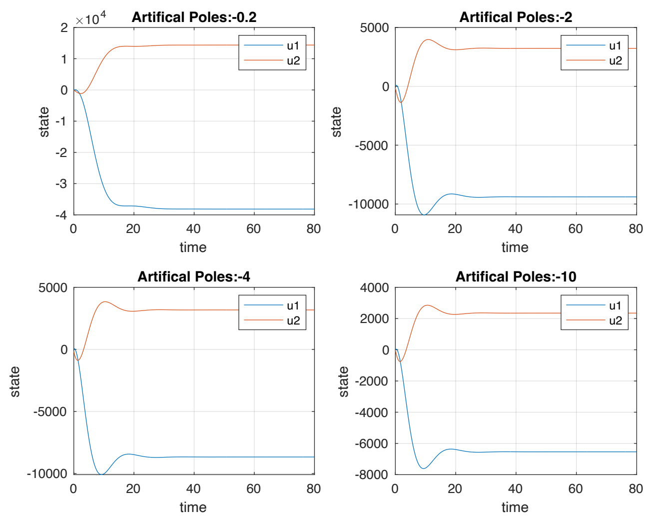


Figure 18 Control signal of step response with different poles

No input

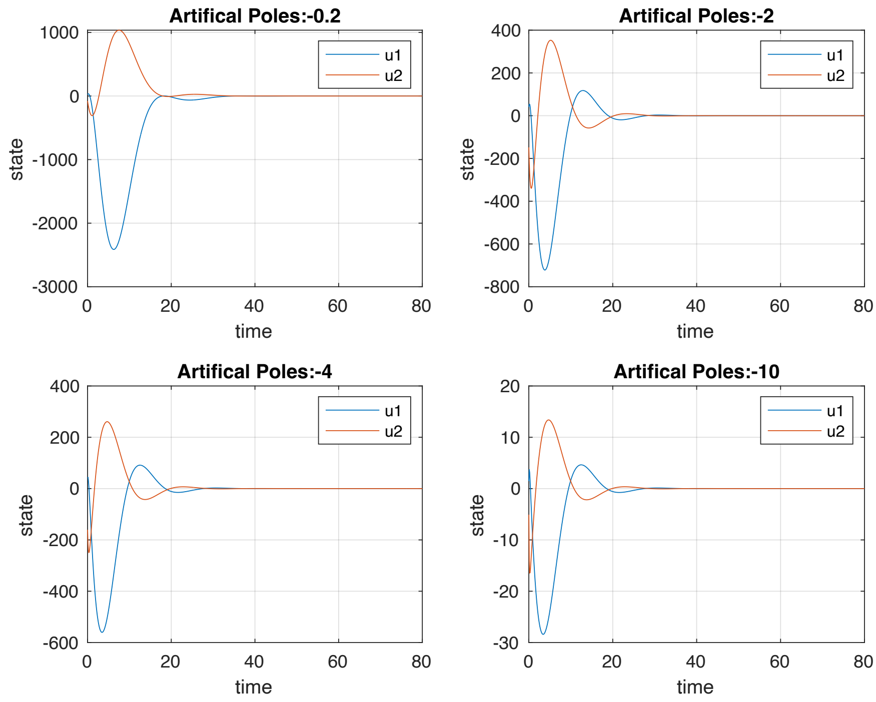


Figure 19 Control signal of non-zero state with different poles

4.3 Discussion

The full rank Unity rank method performs well in meeting the requirements, with its performance being notably influenced by the selection of poles. In scenarios where there is a zero input and non-zero initial states, the states may exhibit temporary fluctuations before converging to zero.

The choice of poles for both full order and unity rank methods significantly impacts system performance. With input, the response of both output and states is notably affected, influencing not only overshoot and settling time but also the steady-state gain. The system's ability to handle high output with small or no input may be compromised.

In the absence of input, different pole selections also have a substantial impact. Larger absolute pole values result in a slight effect on the output, with the inverse being observed in the presence of input.

In summary, the selection of poles is crucial for the system's performance. Achieving a suitable pole selection involves a delicate balance between scenarios with and without input, considering factors such as amplitude fluctuation and system stability.

5. LQR method control

Assume you can only measure the two outputs. Design a state observer, simulate the resultant observer-based LQR control system, monitor the state estimation error, investigate effects of observer poles on state estimation error and closed-loop control performance. In this step, both the disturbance and set point can be assumed to be zero. 15 points

5.1 Derivation procedure

Calculate process：

The LQR optimal control is to find the control law that minimize the cost function:

$$J = \frac{1}{2}\int_0^\infty (x^T Qx + u^T Ru)dt$$

It turns out to be in the form of linear state feedback: u=r-Kx.

First is to find the positive definite solution P of the Riccati quation:

$$PA + A^T P - PBR^{-1}B^T P + Q = 0$$

Where A and B are plant parameters, Q and R are parameters given in LQR method.

It's the key process to solve LQR problem. It will generate many non-linear equations so it's hard to solve it. For singular problem we can directly solve the equation, but for matrix problem here we have systemic steps to solve ARE:

Try

$$Q1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

We have six parameters can be changed.

Step1: Form the matrix

Need MATLAB calculate these numbers:

$$\tau = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix}$$

1	2	3	4	5	6	7	8
-8.8487	-0.0399	-5.0500	3.5846	-0.0042	-2.9260e-04	-0.2959	0.0892
-4.5740	3.7515	-4.3662	-1.0683	-2.9260e-04	-6.7225e-04	-0.0475	0.0181
3.7698	15.9212	-17.2103	4.4936	-0.2959	-0.0475	-21.9696	6.7760
-8.5645	8.3742	-4.4331	-9.2617	0.0892	0.0181	6.7760	-2.1106
-1	0	0	0	8.8487	4.5740	-3.7698	8.5645
0	-2	0	0	0.0399	-3.7515	-15.9212	-8.3742
0	0	-3	0	5.0500	4.3662	17.2103	4.4331
0	0	0	-4	-3.5846	1.0683	-4.4936	9.2617

Eigen values

$$Eigen\ values = \begin{bmatrix} -13.0547 + 8.5766i \\ -13.0547 - 8.5766i \\ 13.0547 + 8.5766i \\ 13.0547 - 8.5766i \\ 7.6806 + 0.0000i \\ -7.6806 + 0.0000i \\ 0.3166 + 0.0000i \\ -0.3166 + 0.0000i \end{bmatrix}$$

There are four stable eigen values

Corresponding vectors are:

1	2	3	4
0.3005 + 0.2583i	0.3005 − 0.2583i	0.2529 + 0.0000i	−0.0214 + 0.0000i
0.1852 + 0.1912i	0.1852 − 0.1912i	−0.1653 + 0.0000i	0.0406 + 0.0000i
0.7499 + 0.0000i	0.7499 + −0.0000i	−0.5339 + 0.0000i	0.0541 + 0.0000i
0.0930 + 0.4161i	0.0930 − 0.4161i	−0.6701 + 0.0000i	0.0245 + 0.0000i
0.0262 − 0.0310i	0.0262 + 0.0310i	0.1674 + 0.0000i	−0.2777 + 0.0000i
0.0387 + 0.1279i	0.0387 − 0.1279i	−0.3684 + 0.0000i	0.9248 + 0.0000i
0.0646 − 0.0045i	0.0646 + 0.0045i	−0.0152 + 0.0000i	−0.0816 + 0.0000i
0.0070 + 0.0652i	0.0070 − 0.0652i	−0.1036 + 0.0000i	−0.2352 + 0.0000i

Step2: Separate the selected stable eigenvectors(corresponding to stable eigenvalues) matrix to two parts:

$$\begin{bmatrix} v_i \\ \mu_i \end{bmatrix}, i = 1, 2, 3, 4$$

μ_i

1	2	3	4
0.3005 + 0.2583i	0.3005 − 0.2583i	0.2529 + 0.0000i	−0.0214 + 0.0000i
0.1852 + 0.1912i	0.1852 − 0.1912i	−0.1653 + 0.0000i	0.0406 + 0.0000i
0.7499 + 0.0000i	0.7499 + −0.00...	−0.5339 + 0.0000i	0.0541 + 0.0000i
0.0930 + 0.4161i	0.0930 − 0.4161i	−0.6701 + 0.0000i	0.0245 + 0.0000i

v_i

1	2	3	4
0.0262 − 0.0310i	0.0262 + 0.0310i	0.1674 + 0.0000i	−0.2777 + 0.0000i
0.0387 + 0.1279i	0.0387 − 0.1279i	−0.3684 + 0.0000i	0.9248 + 0.0000i
0.0646 − 0.0045i	0.0646 + 0.0045i	−0.0152 + 0.0000i	−0.0816 + 0.0000i
0.0070 + 0.0652i	0.0070 − 0.0652i	−0.1036 + 0.0000i	−0.2352 + 0.0000i

Step3: P is given by:

$$Pv_i = u_i, \quad i = 1, 2, 3, 4$$

$$P = \begin{bmatrix} u_1 & u_2 & u_3 & u_4 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}$$

1	2	3	4
2.4115 - 0.0000i	-7.7411 - 0.0000i	0.7335 + 0.0000i	1.9850 + 0.0000i
-7.7411 + 0.0000i	26.0698 + 0.0000i	-2.4312 - 0.0000i	-6.8640 - 0.0000i
0.7335 - 0.0000i	-2.4312 - 0.0000i	0.3118 + 0.0000i	0.6507 + 0.0000i
1.9850 - 0.0000i	-6.8640 - 0.0000i	0.6507 + 0.0000i	2.0780 + 0.0000i

Then we can get the feedback K.

$$K = R^{-1} B^T P$$

1	2	3	4
0.4721 + 0.0000i	-1.1377 + 0.0000i	0.4739 + 0.0000i	-0.0422 - 0.0000i
0.6669 - 0.0000i	-2.1330 - 0.0000i	0.3097 + 0.0000i	0.5001 + 0.0000i

5.2 Simulation results

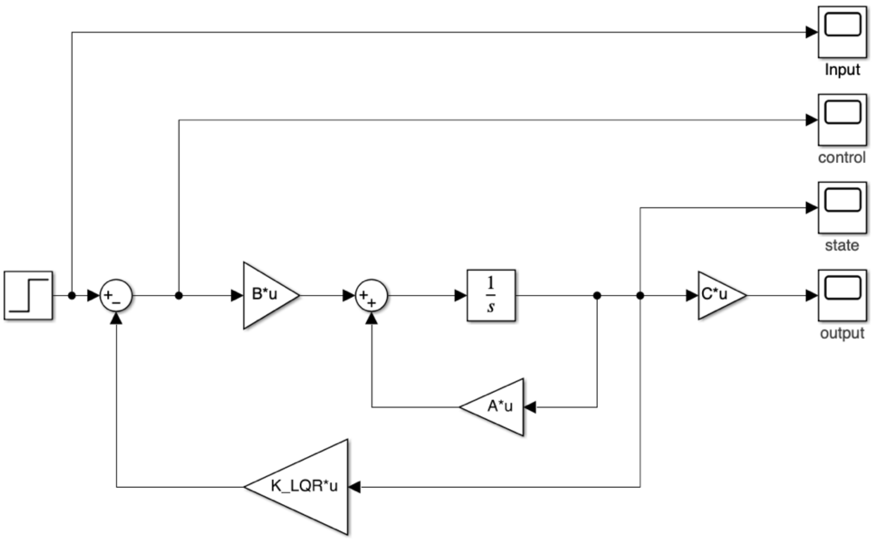


Figure 20 LQR control Simulink structure

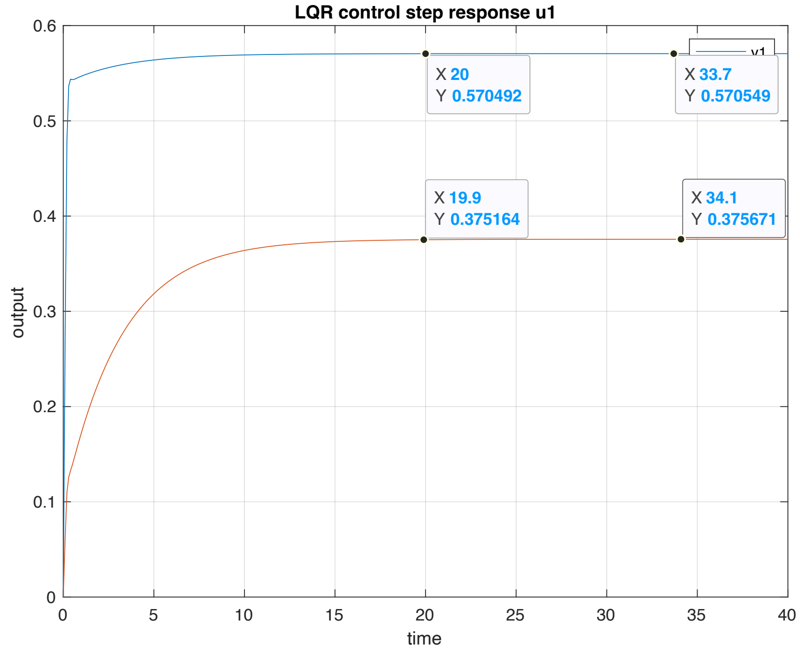


Figure 21 LQR control step response u1

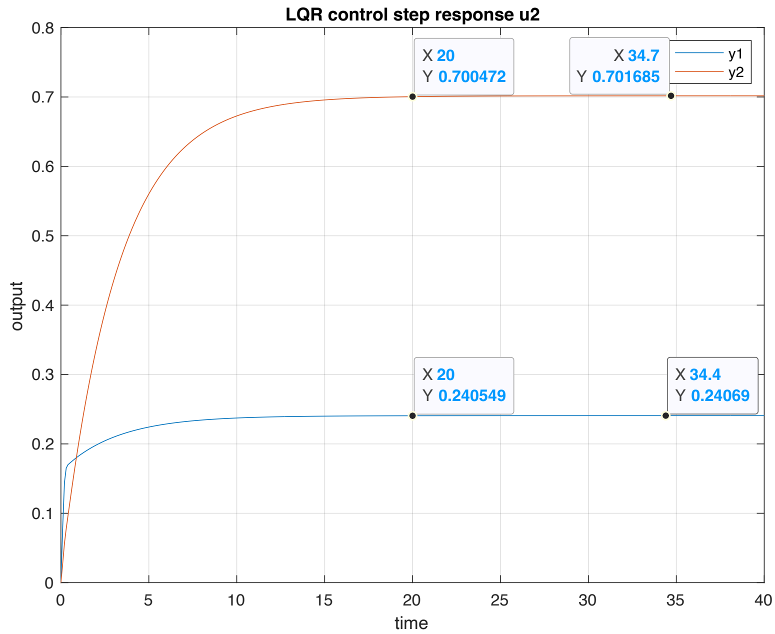


Figure 22 LQR control step response u2

The results satisfy the requirements.

Zero state with step input:

We have six parameters of Q and R can be changed, in order to view the effects of these parameters easily, I keep all the Q in the same and change Q as 0.2, 2, 4, 10 to see the results.

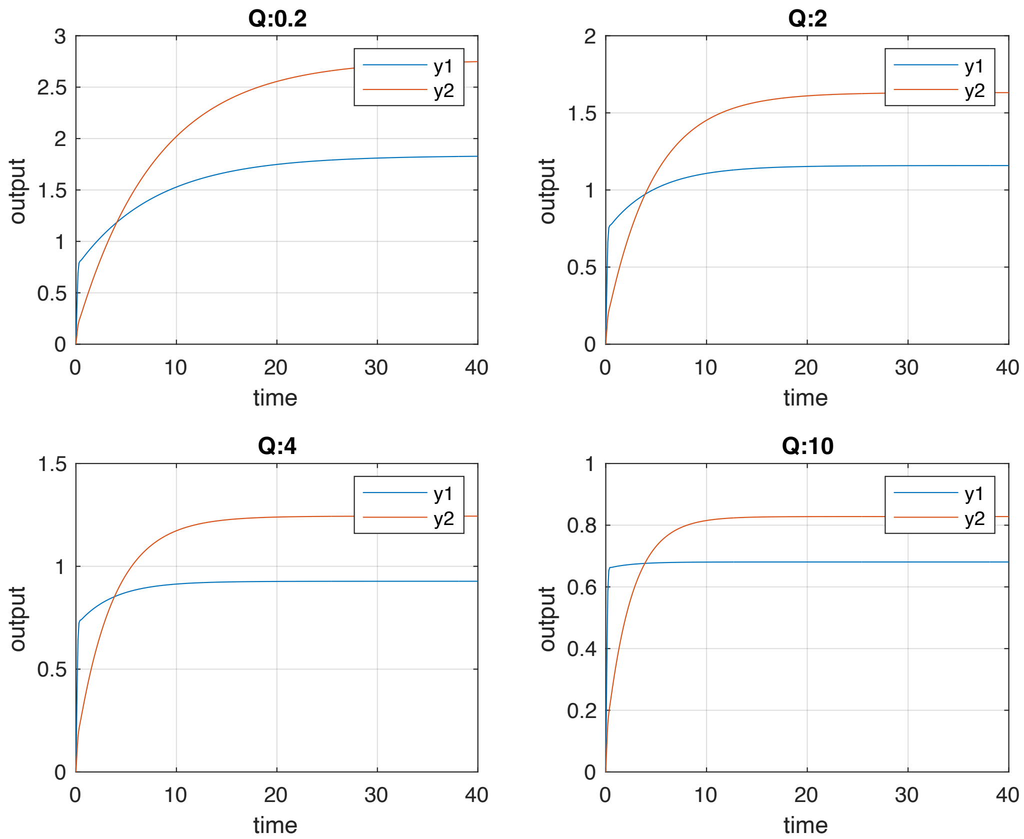


Figure 23 LQR step response with different Q

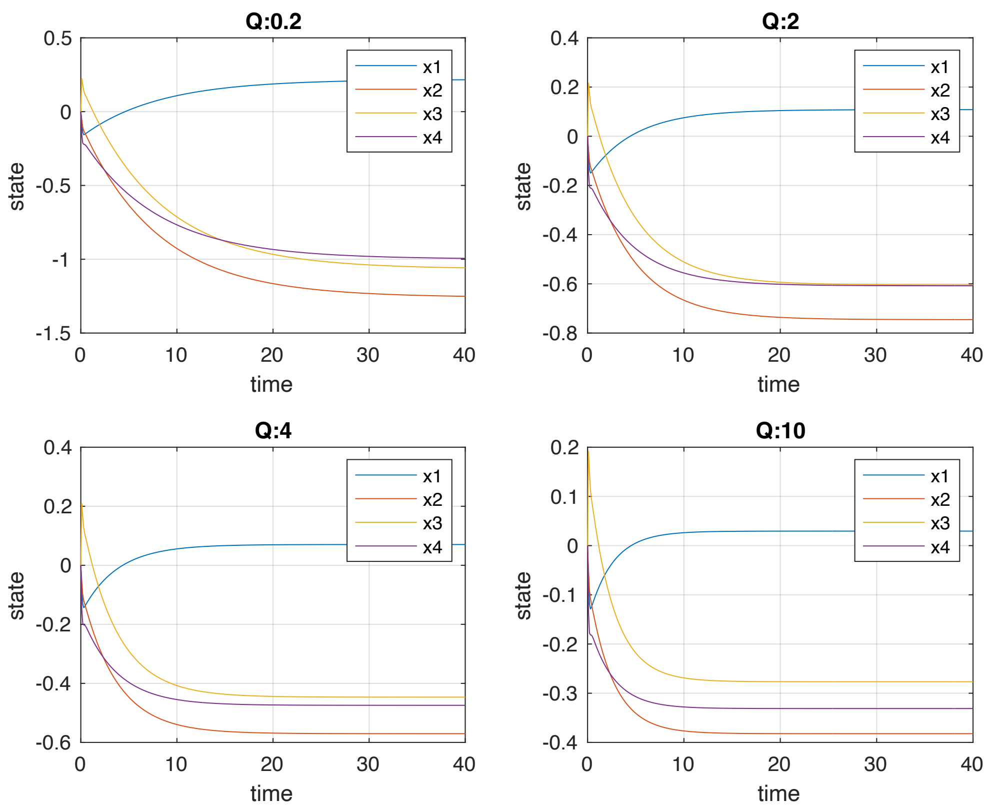


Figure 24 LQR States response with different Q

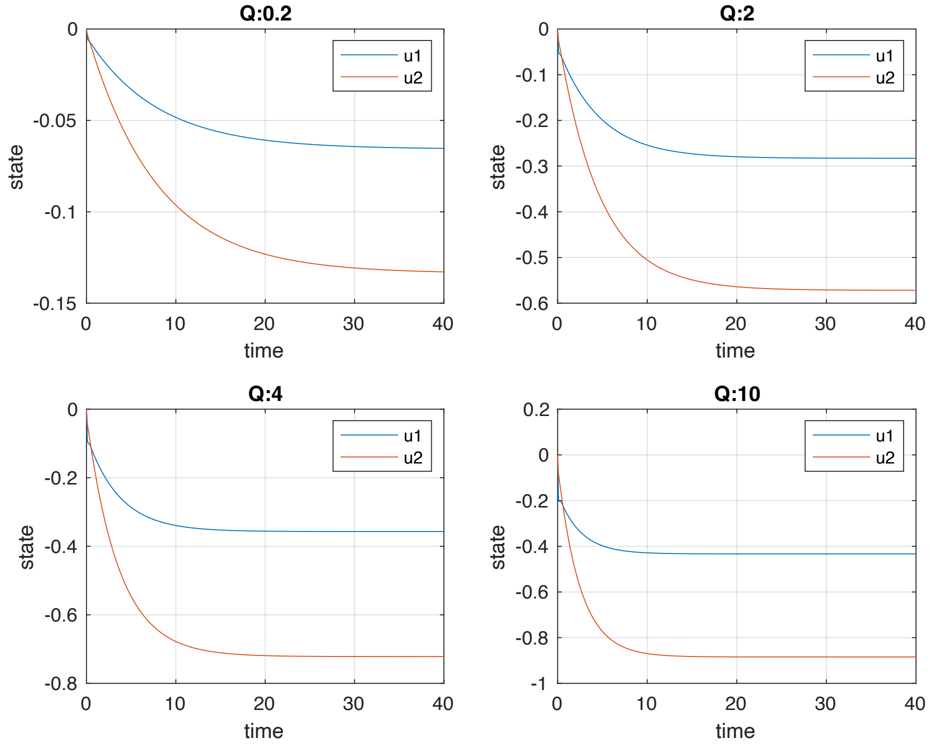


Figure 25 LQR Control signal with different Q

Non zero state with zero input:

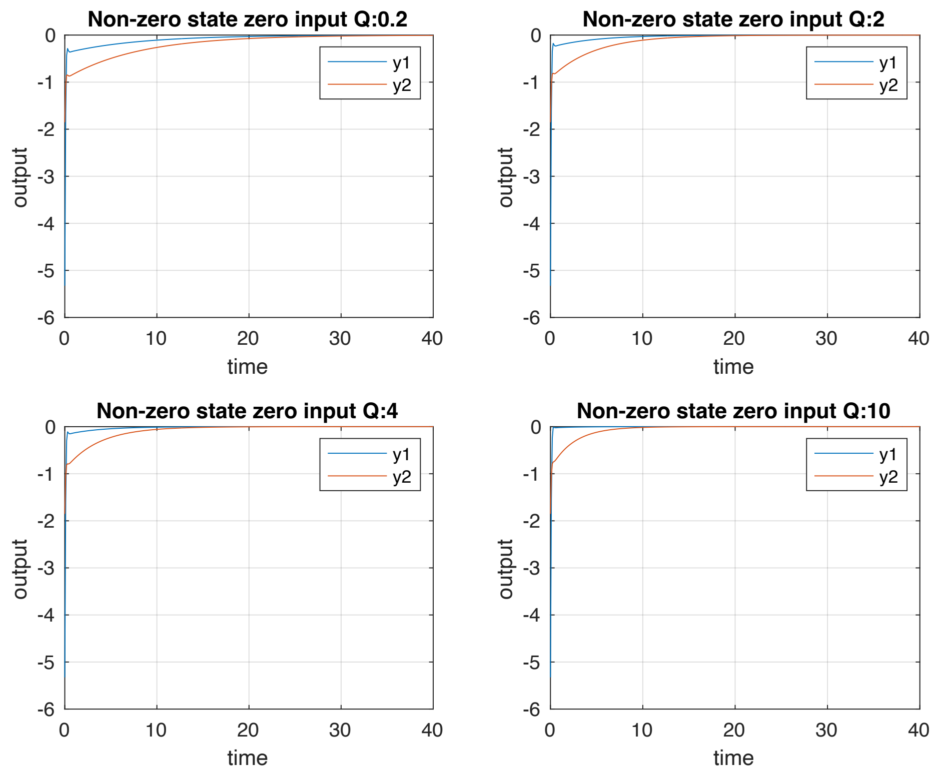


Figure 26 LQR Output of non-zero state with different Q

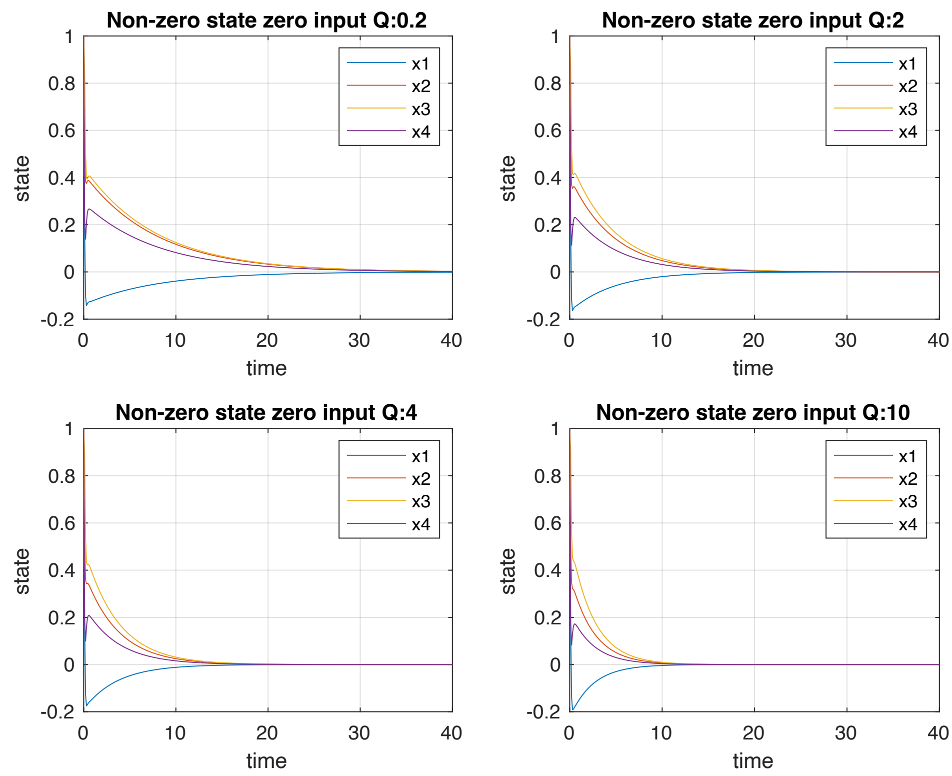


Figure 27 LQR States of non-zero state with different Q

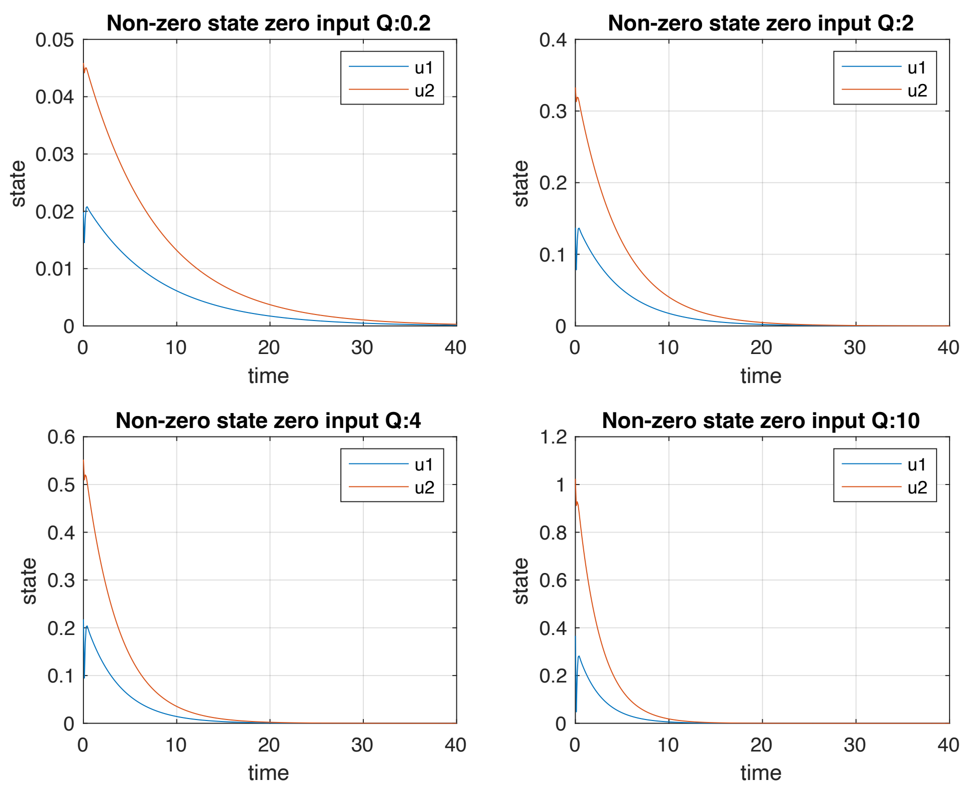


Figure 28 LQR Control signal of non-zero state with different Q

5.3 Discussion

Discuss effects of weightings Q and R on system performance:

The overall performance of the LQR controller in this scenario surpasses that of the pole placement method in the previous question, possibly due to more suitable pole selections in the LQR approach.

In LQR, the Q weight matrix determines the penalty for state convergence speed. A higher Q implies a greater penalty for state settling time. On the other hand, the R weight matrix represents the penalty for input, with a higher R resulting in increased input penalties.

With 4 Q and 2 R adjustable parameters in this case, I've maintained a static R value while simultaneously modifying the four Q parameters for easier analysis. The simulation results indicate that a larger Q leads to faster convergence of steady states, whether with or without input. However, it's important to note that the control signal is also higher with larger Q values. This suggests a need for more power to expedite the cooling of the system, which is reasonable. Thus, achieving a balance between speed and cost involves judiciously choosing different Q and R values.

6. Observer-based LQR control

Assume you can only measure the two outputs. Design a state observer, simulate the resultant observer-based LQR control system, monitor the state estimation error, investigate effects of observer poles on state estimation error and closed-loop control performance. In this step, both the disturbance and set point can be assumed to be zero. (15 points)

6.1 Full order observer

6.1.1 Derivation procedure

Test {A, C} for observability.

$$W_o = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix}$$

1	2	3	4
-3.2988	-2.0532	0.0370	-0.0109
0.2922	-2.1506	-0.0104	0.0163
38.8143	-7.0732	25.0352	-9.3642
7.0724	-8.1087	8.0211	3.1472
-136.5257	292.0883	-554.4793	345.9167
-22.2091	123.3583	-152.3084	40.9094
-5.1808e+03	-4.8300e+03	7.4234e+03	-6.4968e+03
-1.2923e+03	-1.6187e+03	2.0135e+03	-1.2747e+03

Rank(Wo) = 4

Consider an estimator

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L[y = C\hat{x}]$$

Let the estimation error in the state be

$$\tilde{x} = x - \hat{x}$$

$$\dot{\tilde{x}} = \dot{x} - \dot{\hat{x}} = Ax + bu - A\hat{x}(t) + Bu(t) + L[y = C\hat{x}]$$

$$= A(x - \hat{(x)}) - LC(x - \hat{x})$$

$$\tilde{A} = A^T,\ \tilde{B} = C^T\ and \tilde{K} = L^T$$

If we choose L such that (A− LC) = A is stable, we have

$$\dot{\tilde{x}} = (A - LC)\tilde{x} = A_1\tilde{x}$$

$$\tilde{x}(t) = e^{A_1 t}\tilde{x}(0).$$

So need design L to make the estimating system stable so that the estimation error will be zero as time goes to infinity.

Let

$$L = [l_1, l_2, l_3, l_4]$$

$$\det(\lambda I - (A - LC))$$

I set the desired poles are.

$$\begin{bmatrix} -1 & -2 & -3 & -4 \end{bmatrix}$$

So the question now is the same as poles placement in Q1.

The calculation is too complex here, I directly use place function here.

SOD or LQR are both ok to do poles placement here.

6.1.2 Simulation results

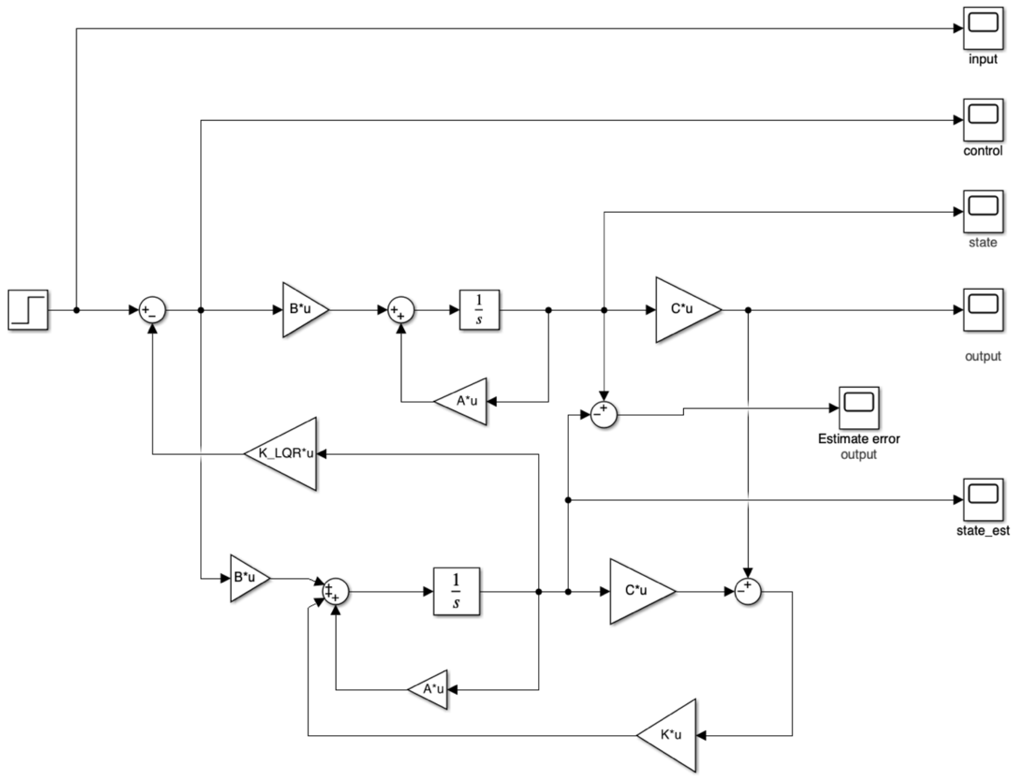


Figure 29 Full order observer with LQR control Simulink structure

$$Q1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix}$$

$$L_{LQR} = \begin{bmatrix} -0.0190 & 0.3604 \\ -0.9050 & -1.5423 \\ -0.6094 & -1.2101 \\ -0.3874 & -0.7230 \end{bmatrix}$$

First, I need check the step response and compare with last part:

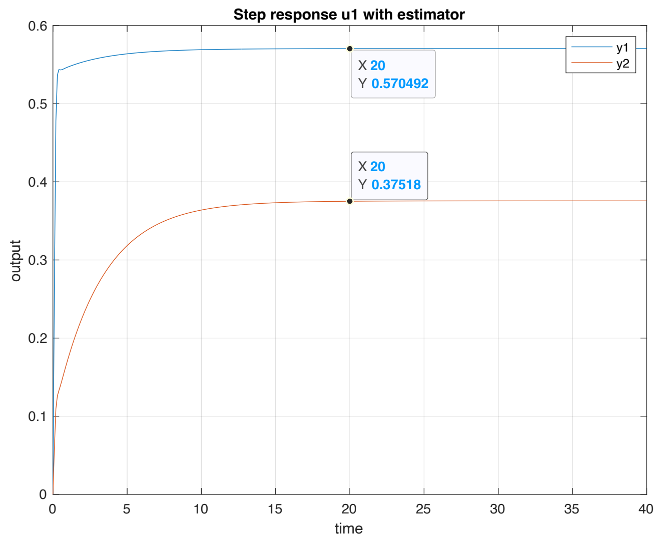


Figure 30 Step response to u1 with estimator

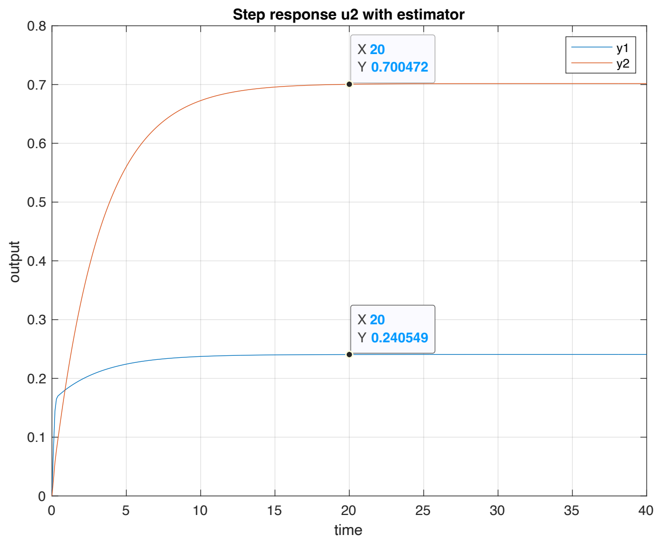


Figure 31 Step response to u2 with estimator

We can see the results are the same as the last part without estimator. That's main because the initial states of the step response system are all zeros, so the estimator can perform well in the beginning and track the states all the time.

Next, check the estimator performance when the initial states are not zero
Estimated error

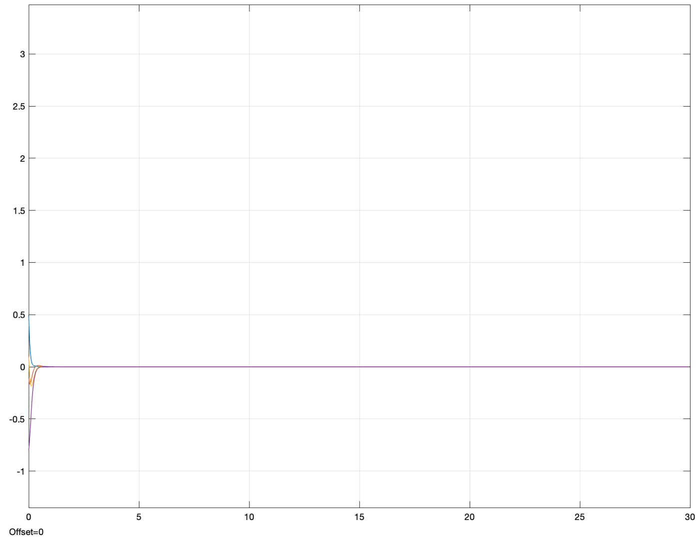


Figure 32 Full order observer estimated error

Estimated states:

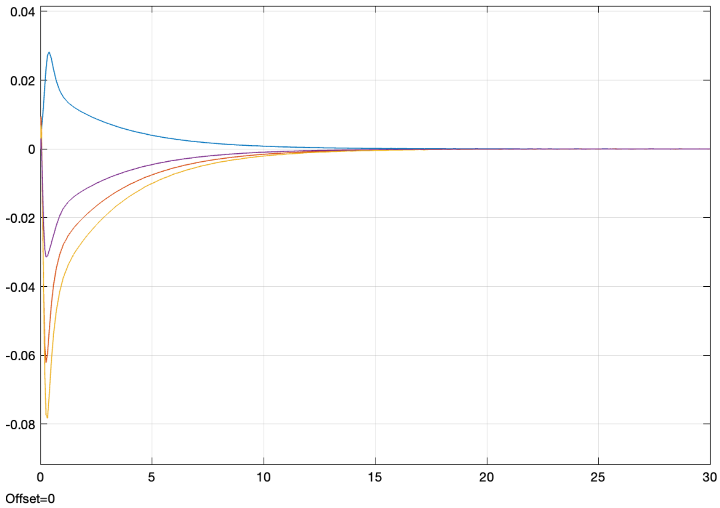


Figure 33 Full order observer estimated states

True states:

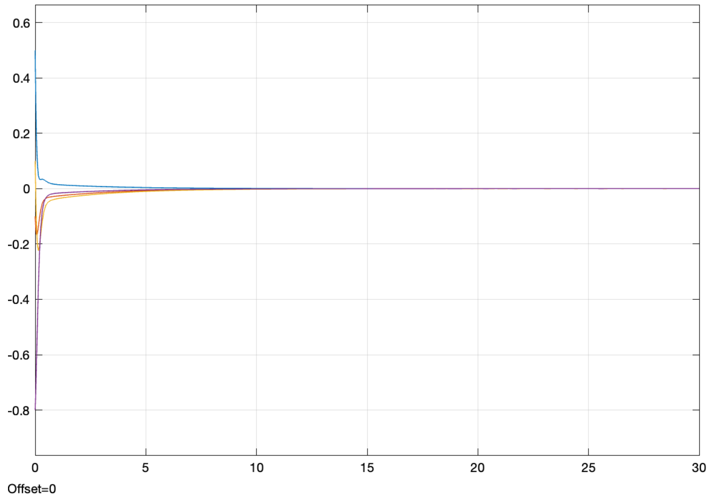


Figure 34 True states

Output:

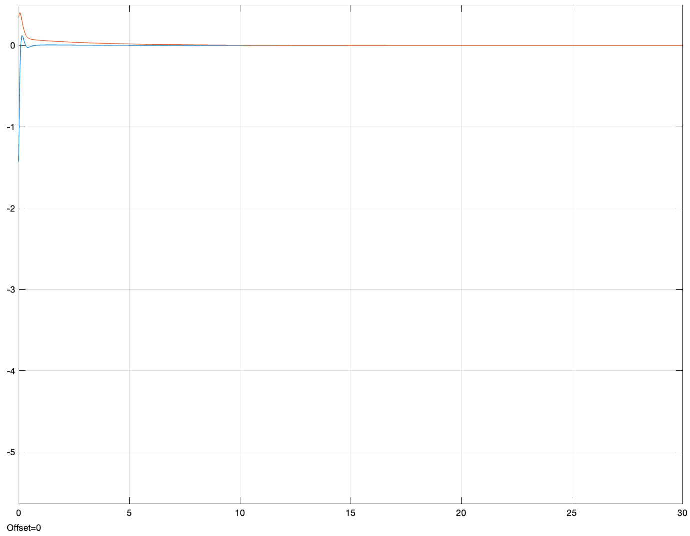


Figure 35 Full order observer with LQR control output

Control signal:

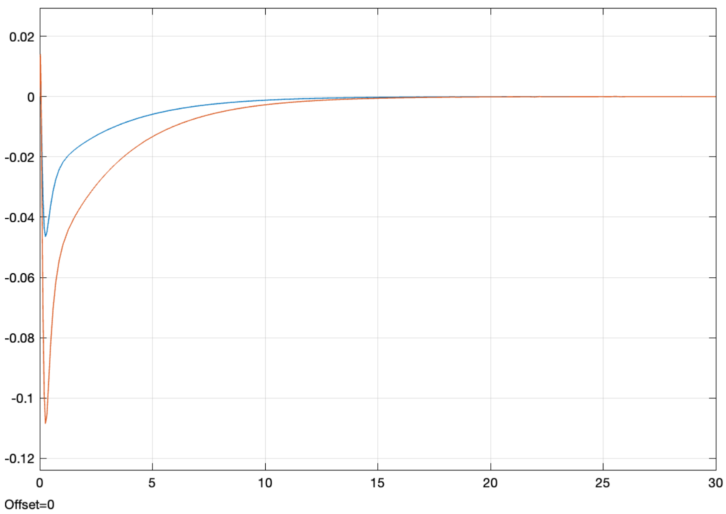


Figure 36 Full order observer LQR control signal

Use poles placement method
Desired poles = [-1, -2, -3, -4];

$$L_{SOD1} = \begin{bmatrix} 5.6827 & -6.5826 \\ 3.1517 & -2.3877 \\ 7.7312 & -11.8073 \\ 2.5571 & 3.3904 \end{bmatrix}$$

Estimated error

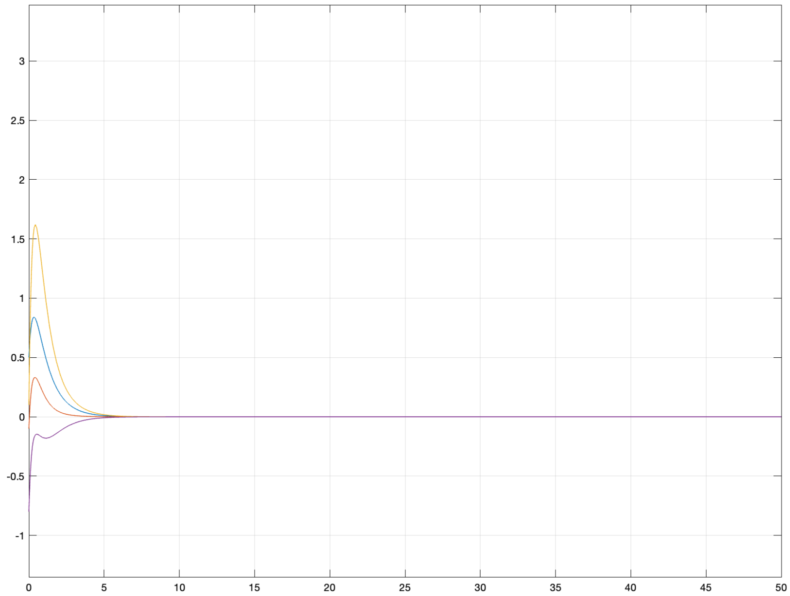


Figure 37 Full order observer estimated error with different poles

Estimated states:

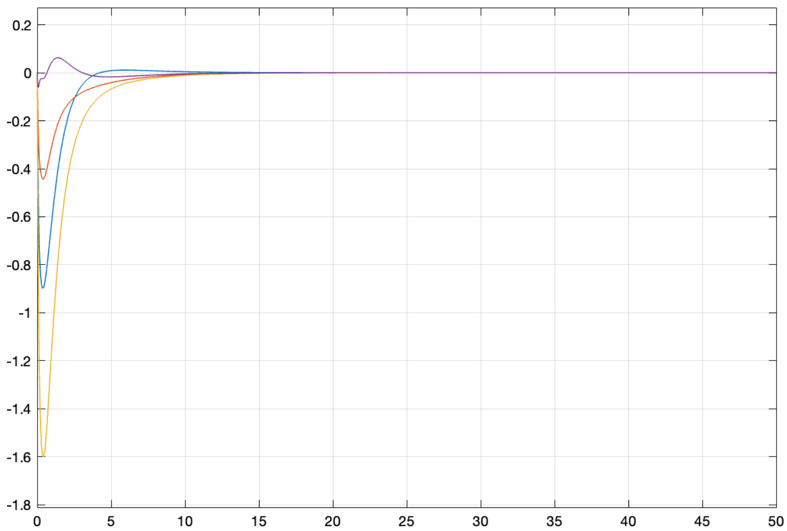


Figure 38 Full order observer estimated states with different poles

True states:

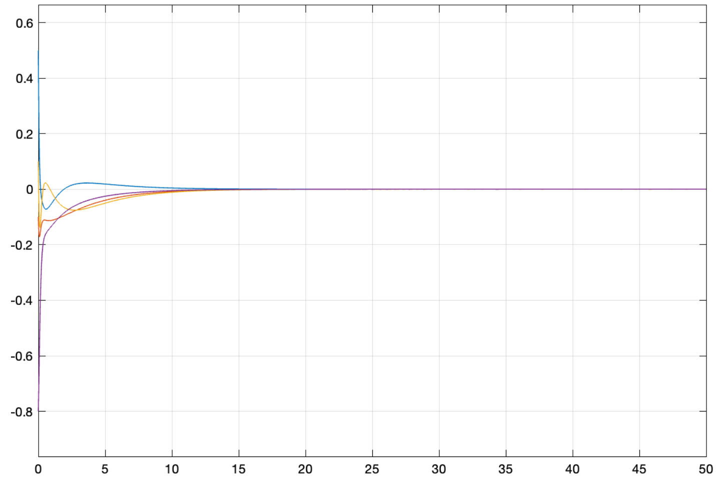


Figure 39 True states

Output:

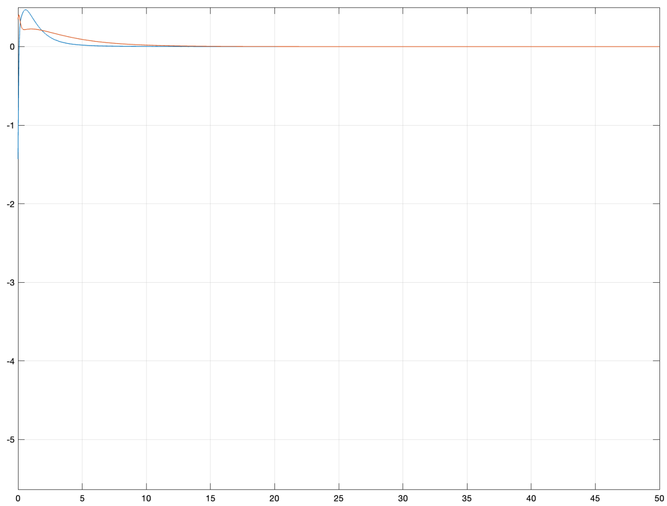


Figure 40 Full order observer with LQR control output under different poles

Control signal:

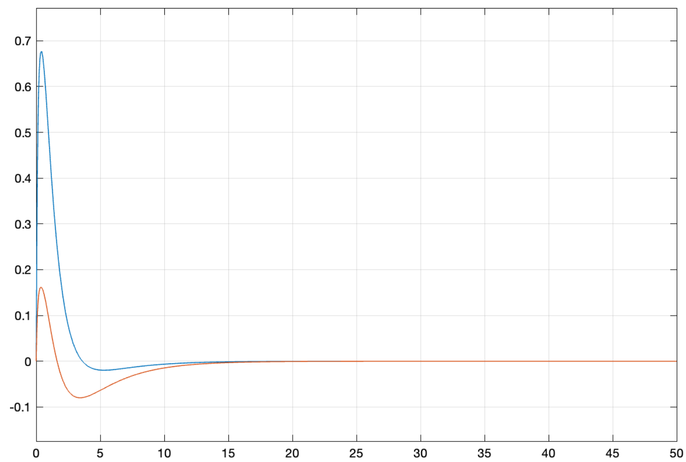


Figure 41 Full order observer LQR control signal with different poles

Desired poles = [-4, -5, -6, -7]

$$L_{SOD2} = \begin{bmatrix} 3.5143 & -3.2514 \\ 3.0086 & -4.1872 \\ 4.0934 & -8.5423 \\ 4.1387 & -3.0015 \end{bmatrix}$$

Estimated error:

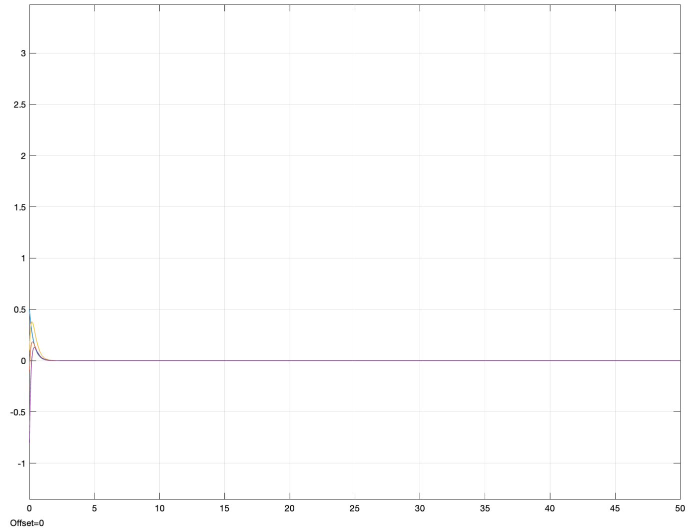


Figure 42 Full order observer estimated error with different poles

Estimated states:

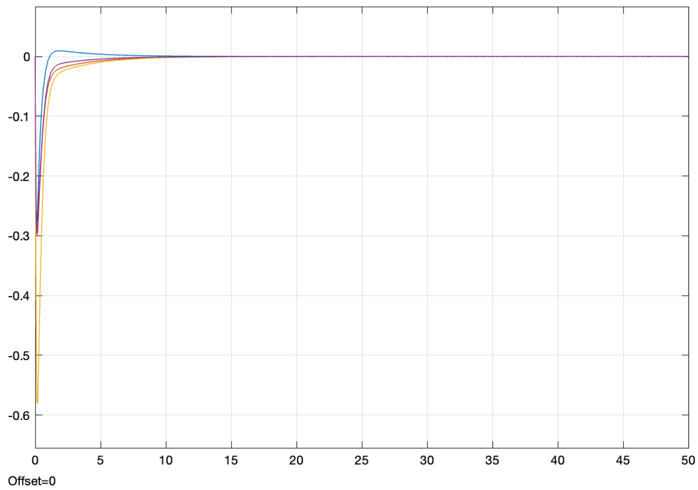


Figure 43 Full order observer estimated states with different poles

True states:

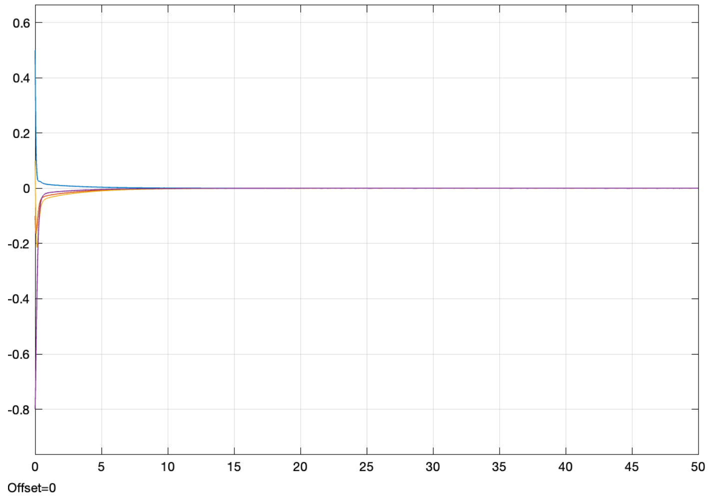


Figure 44 True states

Output:

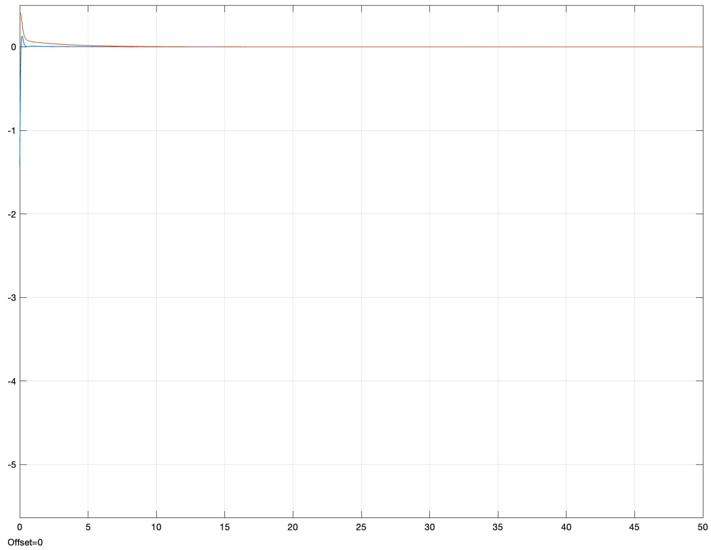


Figure 45 Output of full order observer with LQR control under different poles

Control signal:

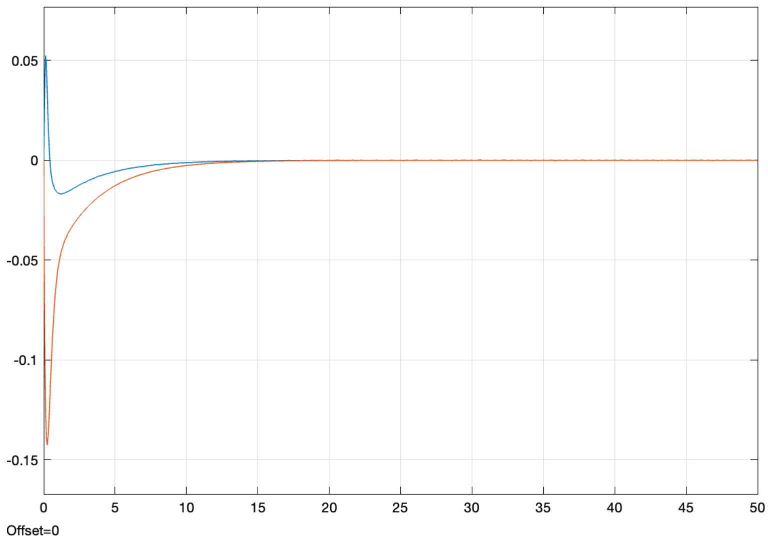


Figure 46 Full order observer LQR control signal under different poles

6.2 Reduced-Order Observers

6.2.1 Derivation procedure

$$\begin{bmatrix} y \\ \xi \end{bmatrix} = \begin{bmatrix} C \\ T \end{bmatrix} x$$

The state can be obtained by inverting

$$x = \begin{bmatrix} C \\ T \end{bmatrix}^{-1} \begin{bmatrix} y \\ \xi \end{bmatrix}$$

Rank(C) = 2
Rank(B) = 4
N = 4> m =2
So we need a second order observer.
We assume:

$$T = \begin{bmatrix} t_1 & t_2 & t_3 & t_4 \\ t_5 & t_6 & t_7 & t_8 \end{bmatrix}$$

Look at

$$\dot{\xi} = D\xi + Eu + Gy$$

$$\xi \to Tx \quad \text{as } t \text{ goes to infinity.}$$

$$[\xi - Tx] \to 0 \quad \text{As } t \to \infty$$

$$\frac{d}{dt}[\xi - Tx] = \dot{\xi} - T\dot{x}$$

$$\dot{\xi} - T\dot{x} = D\xi + Eu + Gy - TAx - TBu$$

$$\frac{d}{dt}[\xi - Tx] = D[\xi - Tx] + (DT - TA + GC)x + (E - TB)u$$

We need the differential equation to be negative and the eigenvalues of D have negative real parts.

$$DT - TA + GC = 0,$$
$$E - TB = 0.$$

I choose.

$$D = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}$$

Solve the equation and then get G, T

```
t1: (2369182511190596876363468400507303417211884396*g1)/3688943799642434449073642484861449017891519811
t2: -(4405800429524699852628503190183851416541322584*g1)/3688943799642434449073642484861449017891519811
t3: (169972600207892430357693845336505398906O608*g1)/3688943799642434449073642484861449017891519811
t4: (1603435556355588687238318095450021326328168448*g1)/3688943799642434449073642484861449017891519811
t5: (2590213678816781543845450513368371168875661854*g2)/5988625773937911862812869366454613349561959811
t6: -(9587067568023877421284934047341630861801189068*g2)/5988625773937911862812869366454613349561959811
t7: (945761299323617188209454196785633416173322240*g2)/5988625773937911862812869366454613349561959811
t8: (3260798076837438005186646202790483830830530560*g2)/5988625773937911862812869366454613349561959811
```

Set G

$$g_1 = 1 \ \ g_2 = 2$$

Get T =

1	2	3	4
0.6422	-1.1943	4.6076e-04	0.4347
0.8650	-3.2018	0.3159	1.0890

$$E = TB$$

1	2
-0.6016	-0.0737
-0.1385	0.2151

Reduced order Observer is

$$\dot{\zeta} = D\zeta + E\mu + Gy = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}\zeta + []\mu + \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} y$$

$$\hat{x} = \begin{bmatrix} C \\ T \end{bmatrix}^{-1} \begin{bmatrix} y \\ \zeta(t) \end{bmatrix} = \begin{bmatrix} -0.2913 & 0.2838 & -0.1273 & 0.0437 \\ -0.0336 & -0.4406 & 0.0423 & -0.0106 \\ -0.7109 & 0.377 & -0.8232 & 3.2732 \\ 0.3389 & -1.6302 & 2.6135 & -0.0972 \end{bmatrix} \begin{bmatrix} y \\ \zeta(t) \end{bmatrix}$$

6.2.1 Simulation results

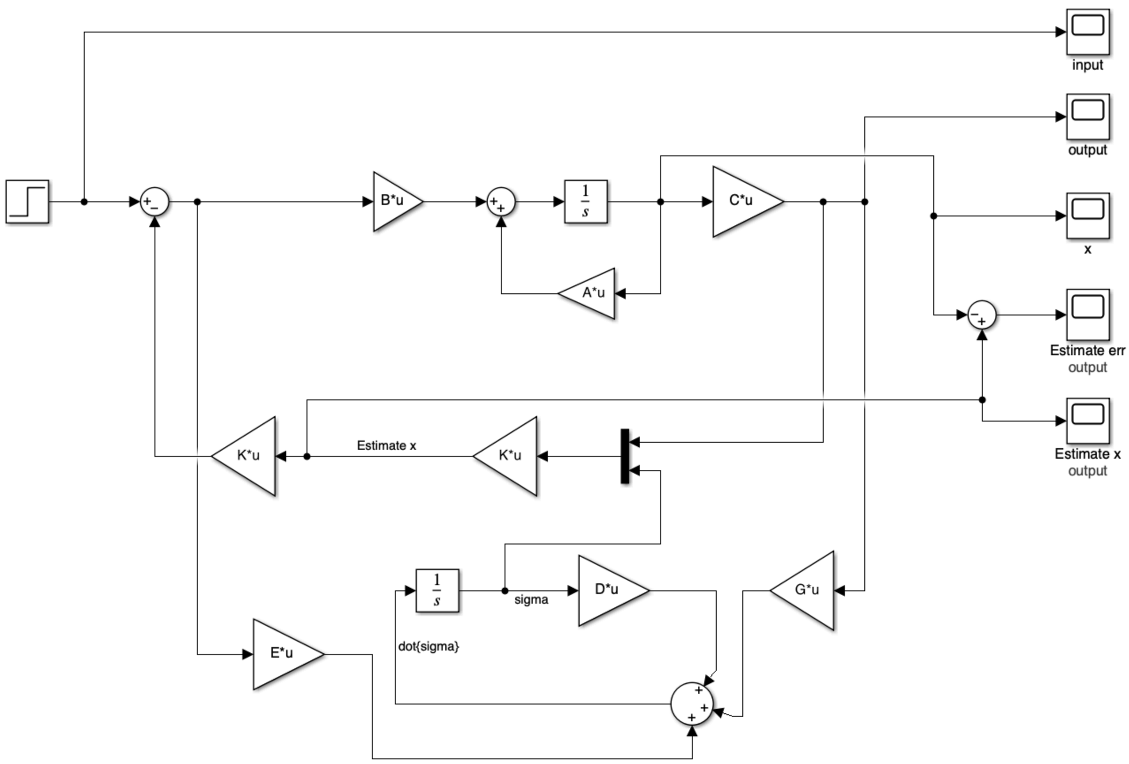


Figure 47 Reduced order Observer Simulink structure

Estimated error:

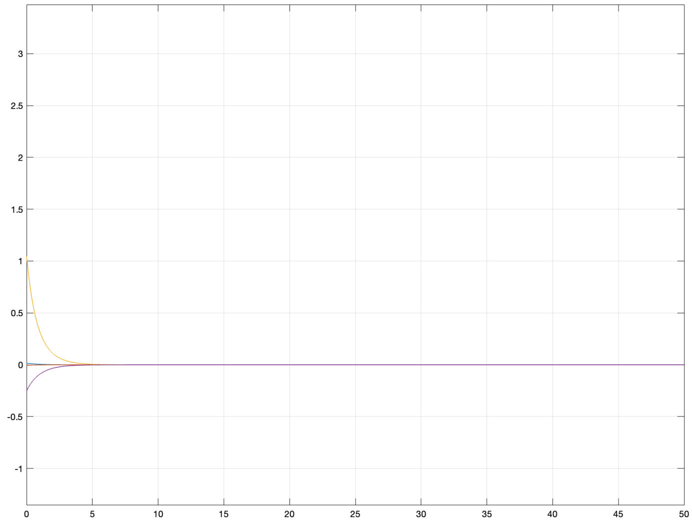


Figure 48 Reduced order Observer estimated error

Estimated states:

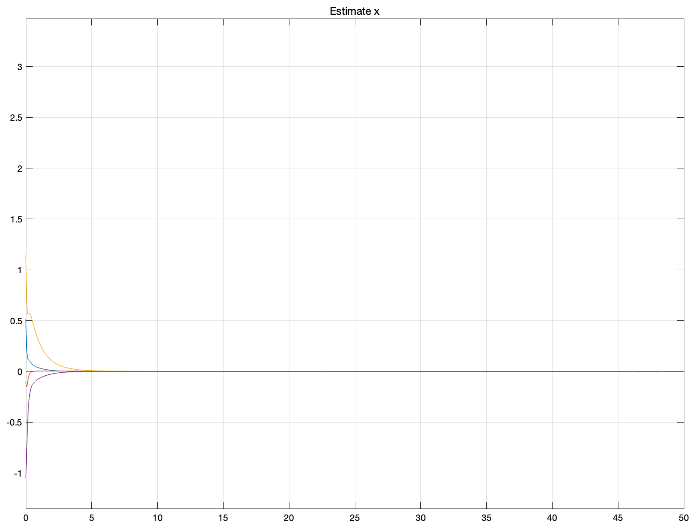


Figure 49 Reduced order Observer estimated states

Ture states:

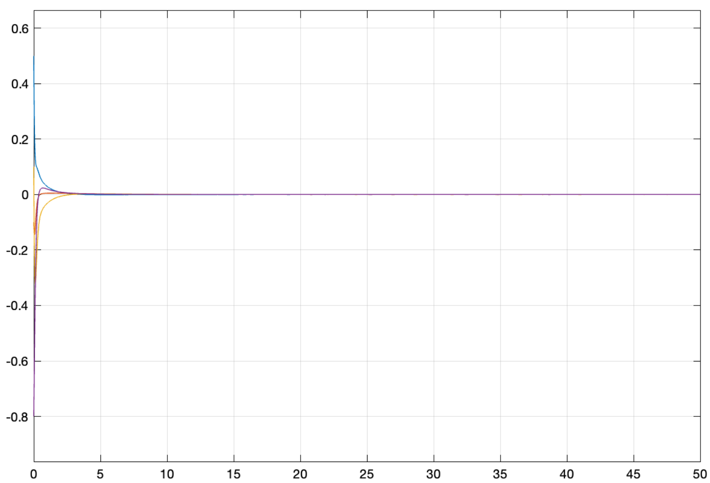


Figure 50 True states

Output:

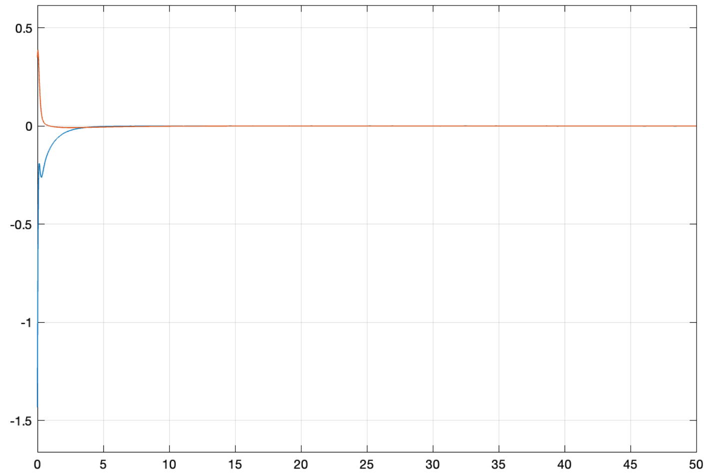


Figure 51 Reduced order Observer output with LQR control

Control input:

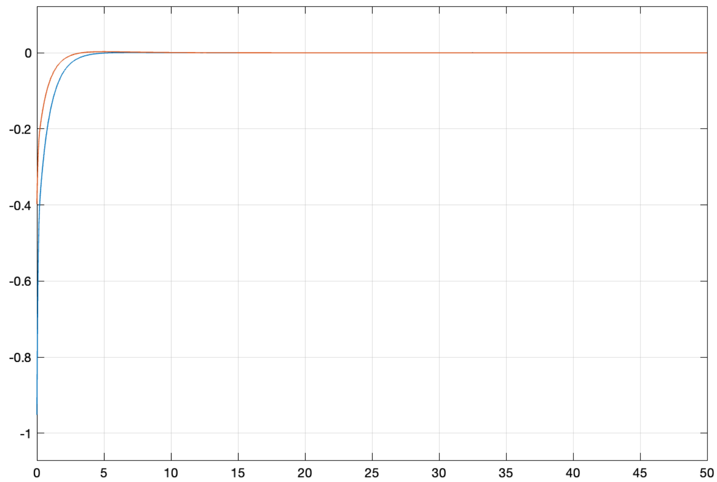


Figure 52 Reduced order Observer LQR control input

The estimator input:

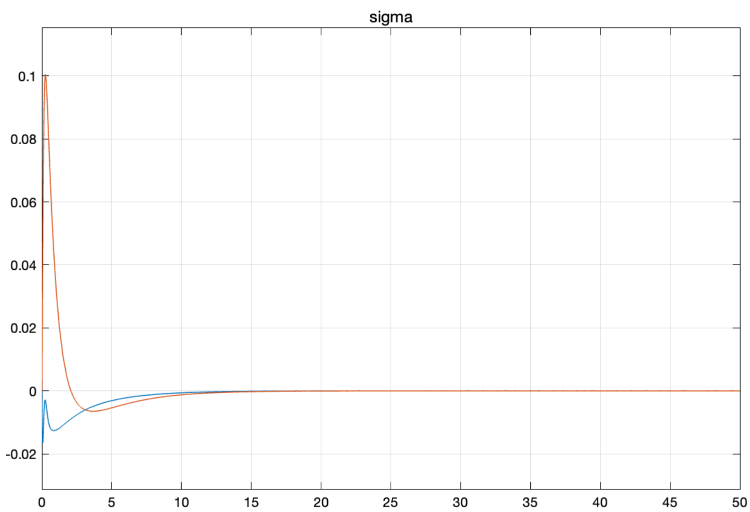


Figure 53 The estimator input

D = [-5 0;
0 -8];

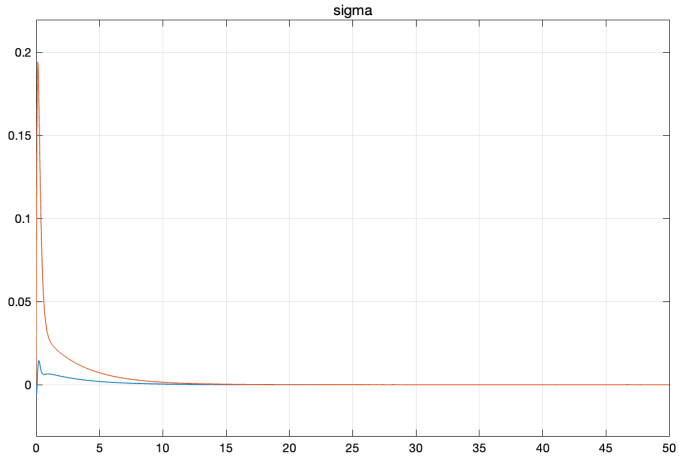


Figure 54 The estimator input with different poles D

6.3 Discussion

In this particular scenario, I employed both full-order and reduced-order observers for system observation while maintaining LQR as the external controller, akin to the feedback used in the previous question.

The simulation results unequivocally indicate that the observer's performance is highly effective. The selection of different poles for the observer not only impacts the speed at which the estimation error converges to zero but also affects the system's output. Faster convergence of the estimated error corresponds to a quicker convergence of the output. However, it's crucial to note that there's no free lunch – the controller input for the estimator is higher when the absolute value of the pole, particularly in the reduced-order observer, is larger. This observation underscores the trade-offs involved in choosing observer poles, considering both estimation accuracy and control effort.

7. Decoupling controller

Design a decoupling controller with closed-loop stability and simulate the step response of the resultant control system to verify decoupling performance with stability. In this question, the disturbance can be assumed to be zero. Is the decoupled system internally stable? Please provide both the step (transient) response with zero initial states and the initial response with respect to xx of the decoupled system to support your conclusion. (20 points)

7.1 State decoupling controller

The decoupling control is:

$$\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x} + \boldsymbol{F}\boldsymbol{r}$$

To calculate sigma.

$$c_1^T \boldsymbol{B} = [-0.0381 - 0.0119]$$
$$c_2^T \boldsymbol{B} = [-0.08900.0340]$$
$$\sigma_1 = 1$$
$$\sigma_2 = 1$$

In this question, sigma 1 and sigma 2 are both equal to 1.

Select two poles are -2 -3, desired G(s) is

$$G(s) = \begin{bmatrix} \frac{1}{A+2I} & 0 \\ 0 & \frac{1}{A+3I} \end{bmatrix}$$

$$C^\star = G(s) = \begin{bmatrix} c_1^T A^{\sigma_1} \\ c_2^T A^{\sigma_2} \end{bmatrix}$$

$$B^\star = G(s) = \begin{bmatrix} c_1^T A^{\sigma_1 - 1} B \\ c_2^T A^{\sigma_2 - 1} B \end{bmatrix}$$

C star

1	2	3	4
32.2167	-11.1796	25.1092	-9.3860
7.9490	-14.5605	7.9899	3.1961

B star

1	2
-0.0381	-0.0119
-0.0890	0.0340

$$F = (B^{\star})^{-1} \quad K = (B^{\star})^{-1} \begin{bmatrix} c_1^T \phi_{f1}(A) \\ c_2^T \phi_{f2}(A) \end{bmatrix}$$

F:

1	2
-14.4429	-5.0527
-37.7733	16.1715

K:

1	2	3	4
-505.4649	235.0346	-403.0185	119.4121
-1.0884e...	186.8234	-819.2488	406.2277

Closed feedback system controllability matrix Wc:

1	2	3	4	5	6	7	8
-2.0195	0.2309	-5.6662e+03	-75.6548	-1.8385e+07	-2.4347e+05	-5.9645e+10	-7.8990e+08
0.5172	-0.4068	1.8203e+03	25.0106	5.9049e+06	7.8206e+04	1.9157e+10	2.5371e+08
-115.2220	-1.1645	-3.7382e+05	-4.9749e+03	-1.2128e+09	-1.6061e+07	-3.9347e+12	-5.2108e+10
30.9206	2.7949	1.0323e+05	1.2979e+03	3.3485e+08	4.4362e+06	1.0864e+12	1.4387e+10

Rank(Wc) = 3

It’s not controllable anymore.

I simulate it many time and found the result is not stable.

For this question, I tried many times but can’t decouple successfully.

7.2 Decoupling Control by Output Feedback

closed-loop transfer function matrix:

$$H(s) = [I + G(s)K(s)]^{-1}G(s)K(s)$$

It is clear that the closed loop H is decoupled if the open loop GK is so.

So we need decouple

$$G(s)K(s)$$

designing $K(s)$ in two stages, i.e., $K(s) = K_d(s)K_s(s)$

$K_d(s)$ is to make $G(s)K_d(s)$ diagonal and non-singular. In this step, $K_d(s)$ is not required to be proper.

$$G(s) = \frac{N(s)}{d(s)}$$

polynomial is very complex, it is shown in the appendix page.
Suppose I can get least common denominator d(s)
Next I need

$$det\ N(s)$$

To make sure N(s) and d(s) have no common unstable roots.
Next can choose

$$K_d = adjN(s)$$

Kd may not proper

We need design a diagonal controller to K_s stabilize the decoupled system and make $K(s) = K_d(s)K_s(s)$ proper.
During
the whole process, need to make sure no unstable pole-zero cancellation between G(s) and K(s)

7.3 Simulation results

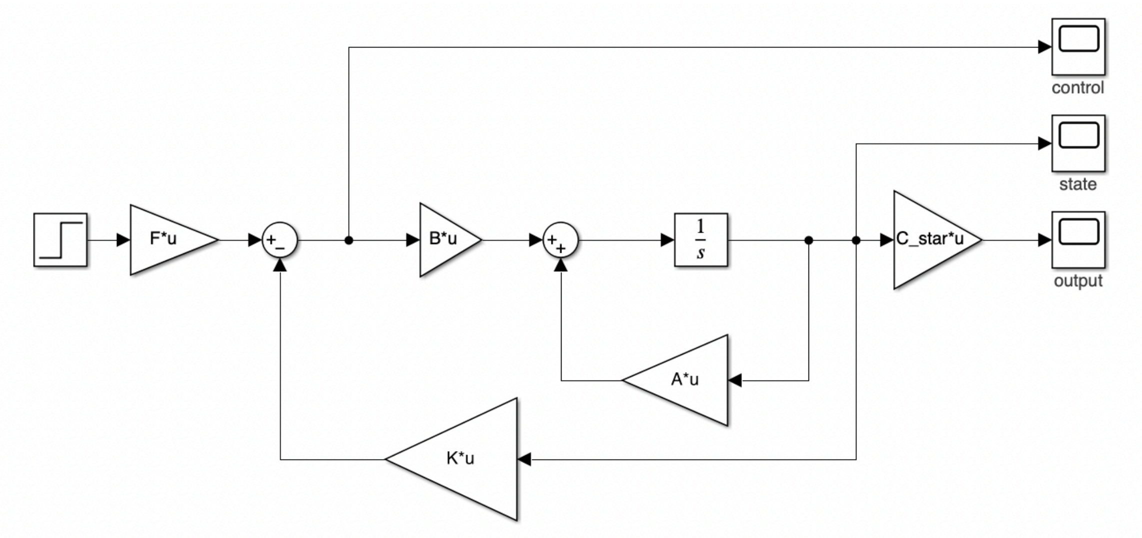


Figure 55 State decupling Simulink structure

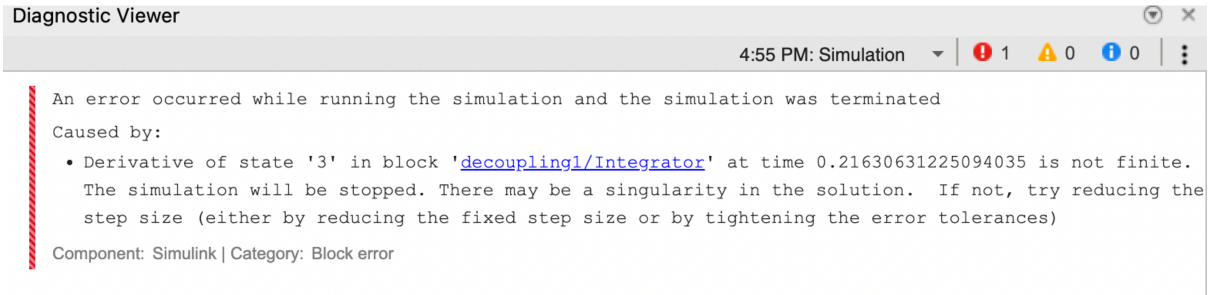


Figure 56 Simulink Decoupling Error

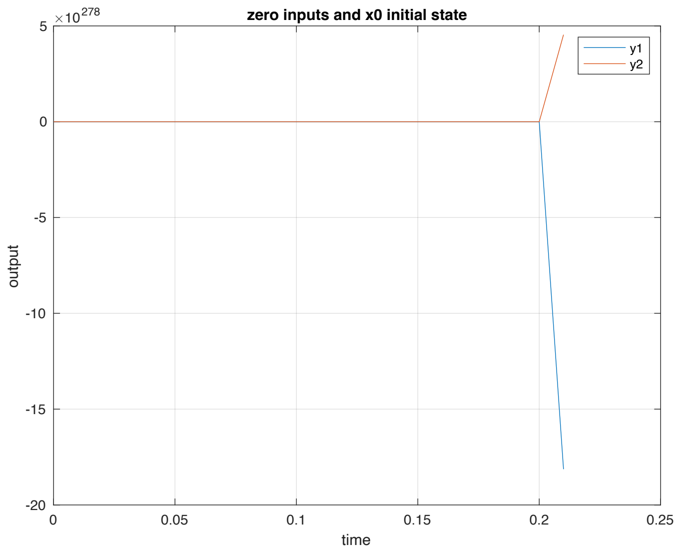


Figure 57 Simulation output of m file

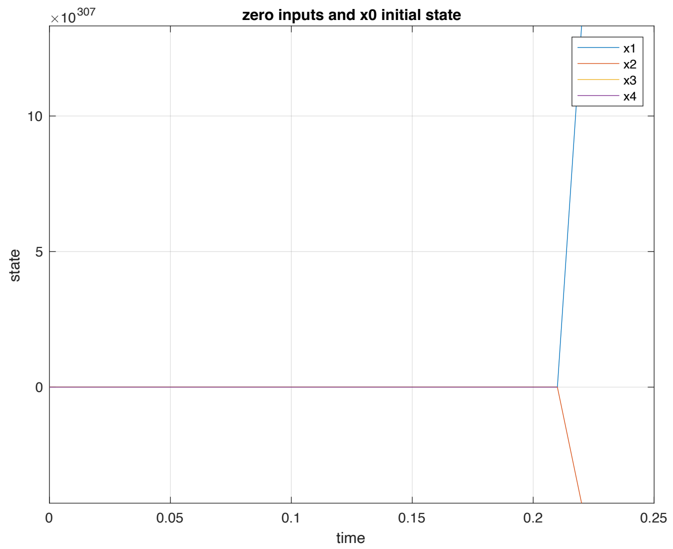


Figure 58 Simulation states results of m file

7.4 Discussion

In tackling this issue, I experimented with two methods. Regrettably, the first method proved ineffective; although the system exhibited decouplable, the results were unstable. I attempted a secondary approach by using output for decoupling, but the calculations proved to be impractical, as detailed earlier.

Significant time was devoted to uncovering the root cause of the instability. I suspect that residual coupling effects between subsystems were not fully considered, giving rise to interactions among different components of the system and ultimately contributing to instability.

Furthermore, I discovered that the decoupled system I derived is no longer controllable. This newfound lack of controllability might be an additional factor hindering the attainment of stable states.

8. Servo controller

In an application, the operating set point for the two outputs is y_{sp} =[0.4,0.8]^T. Assume that you only have two sensors to measure the output. Design a controller such that the plant (the diesel engine system) can operate around the set point as close as possible at steady state even when step disturbances are present at the plant input. Plot out both the control and output signals. In your simulation, you may assume the step distur

8.1 Derivation procedure

This question is a MIMO system, so we can apply Multivariable Integral Control.

$$\dot{x} = Ax + Bu + B_w w$$
$$y = Cx$$

The error vector is

$$e = r - y$$

The objective is to achieve zero steady state error for all the outputs.

Let the state feedback controller be

$$u = -Kx + Fr$$

$$\dot{x} = Ax + Bu + B_w w = (A - BK)x + BFr + B_w w$$

The disturbance *w* and reference *r* are **both of consistent**. To achieve zero steady state error, like SISO integral control, we introduce one integrator to each channel.

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} A & O \\ -C & O \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} B \\ O \end{bmatrix} u + \begin{bmatrix} B_w \\ O \end{bmatrix} w + \begin{bmatrix} 0 \\ I \end{bmatrix} r$$

$$\dot{\bar{x}} = \bar{A}\bar{x} + \bar{B}_w w + \bar{B}_r r$$

$$y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} = \bar{C}\bar{x}$$

This system order is n = 4. Input m = 2

So we need check the system is controllable or not.

Wc =

1	2	3	4	5	6
−8.8487	−0.0399	−5.0500	3.5846	0.0564	0.0319
−4.5740	3.7515	−4.3662	−1.0683	0.0165	−0.0200
3.7698	15.9212	−17.2103	4.4936	4.4939	1.3321
−8.5645	8.3742	−4.4331	−9.2617	−1.4269	−0.2730
−3.2988	−2.0532	0.0370	−0.0109	0	0
0.2922	−2.1506	−0.0104	0.0163	0	0

rank(Wc) = n+m = 6

Then set controoler:

$$u = -K\bar{x} = -\begin{bmatrix} K_1 & K_2 \end{bmatrix}\begin{bmatrix} x \\ v \end{bmatrix}$$

The resultant feedback system is

$$\bar{A} = \begin{bmatrix} -8.8487 & -0.0399 & -5.0500 & 3.5846 & 0 & 0 \\ -4.5740 & 3.7515 & -4.3662 & -1.0683 & 0 & 0 \\ 3.7698 & 15.9212 & -17.2103 & 4.4936 & 0 & 0 \\ -8.5645 & 8.3742 & -4.4331 & -9.2617 & 0 & 0 \\ 3.2988 & 2.0532 & -0.0370 & 0.0109 & 0 & 0 \\ -0.2922 & 2.1506 & 0.0104 & -0.0163 & 0 & 0 \end{bmatrix}$$

$$\bar{B} = \begin{bmatrix} 0.0564 & 0.0319 \\ 0.0165 & -0.0200 \\ 4.4939 & 1.3321 \\ -1.4269 & -0.2730 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\bar{B}_w = \begin{bmatrix} 0.0564 & 0.0319 \\ 0.0165 & -0.0200 \\ 4.4939 & 1.3321 \\ -1.4269 & -0.2730 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\bar{B}_r = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\bar{C}_r = \begin{bmatrix} -3.2988 & -2.0532 & 0.0370 & -0.0109 & 0 & 0 \\ 0.2922 & -2.1506 & -0.0104 & 0.0163 & 0 & 0 \end{bmatrix}$$

Now the problem has been changed into poles placement problem, the solutions are the same with mentioned methods, I can use LQR methods.

8.2 Simulation results

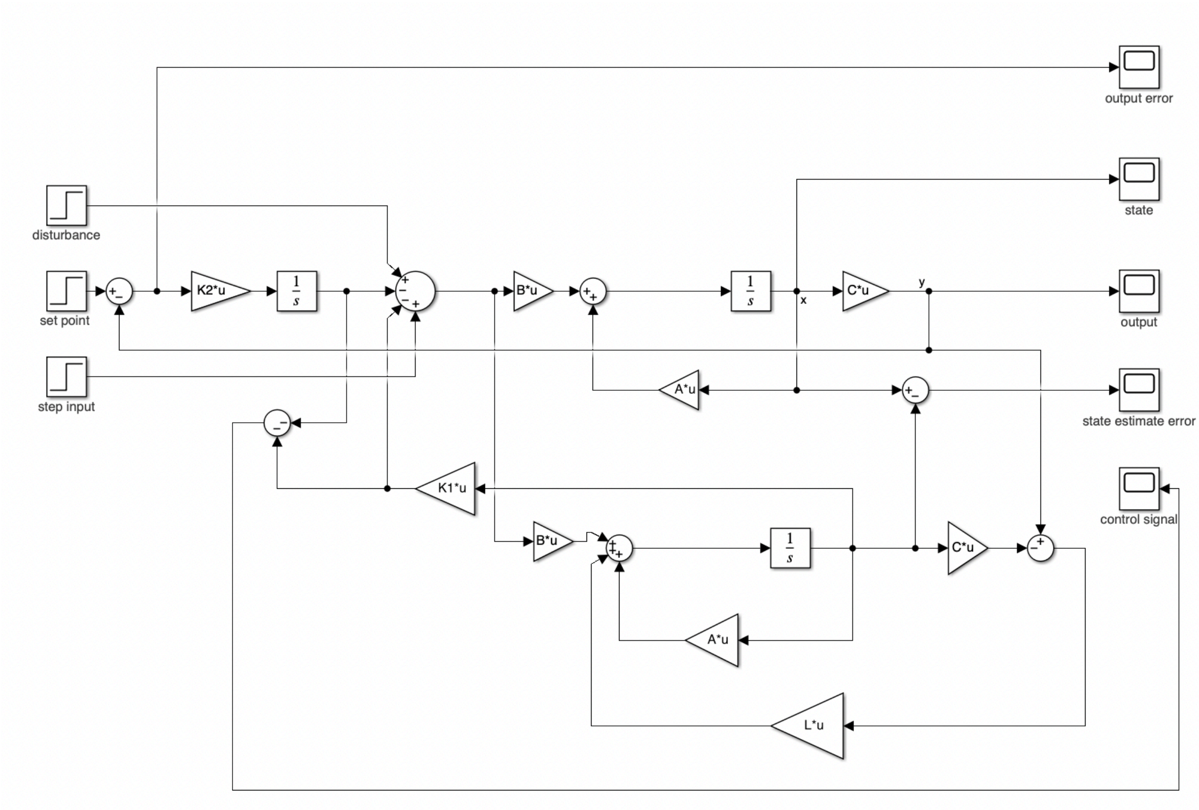


Figure 59 Servo control Simulink structure

The step disturbance is added in 10s.

The output:

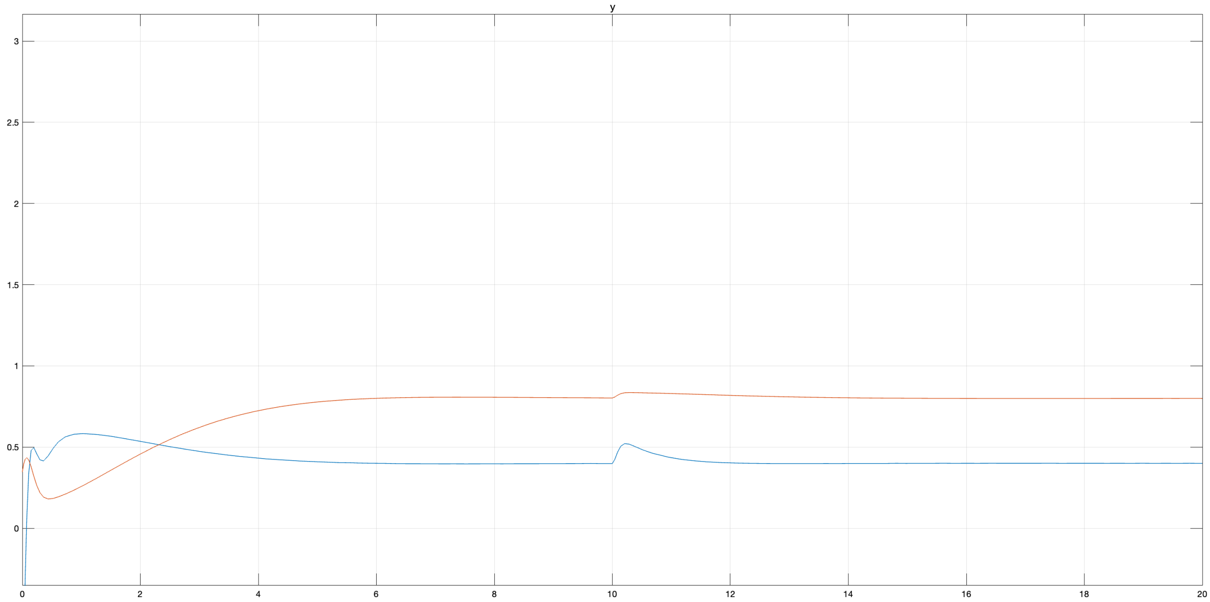


Figure 60 Output of Servo control

State estimate error

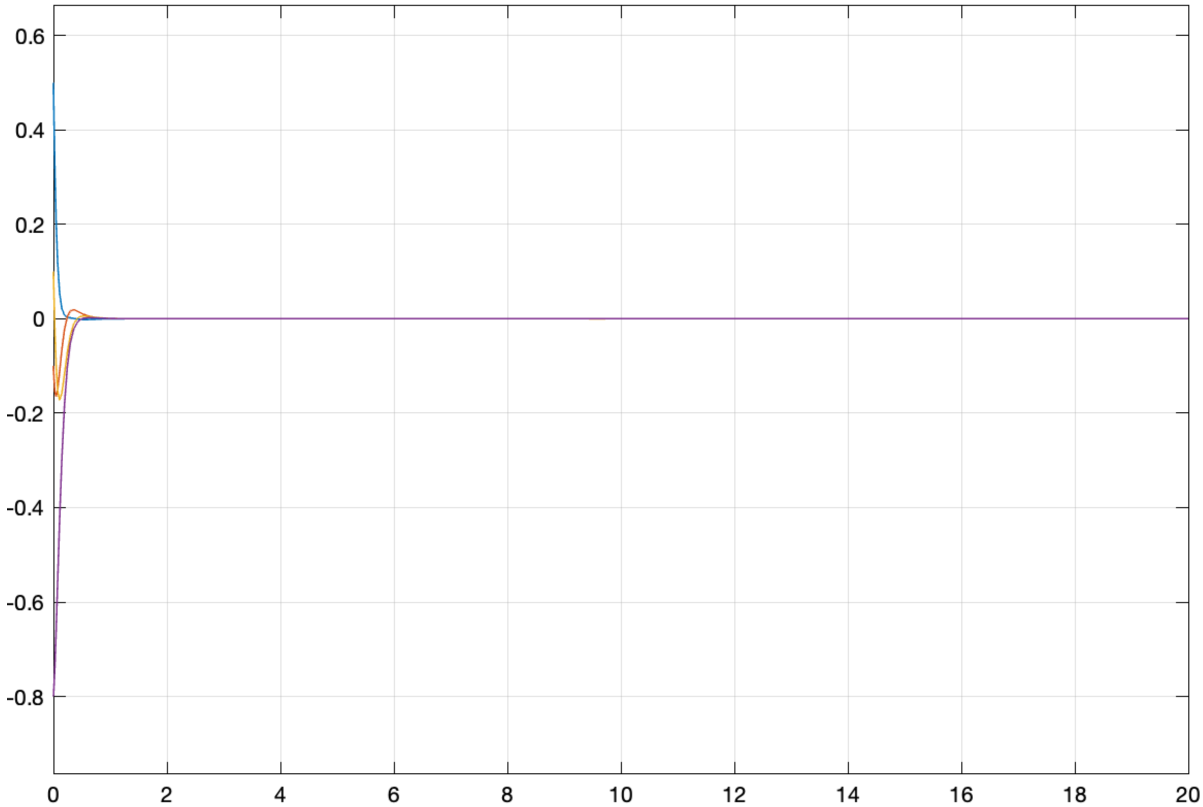


Figure 61 State estimated error of Servo control

True state:

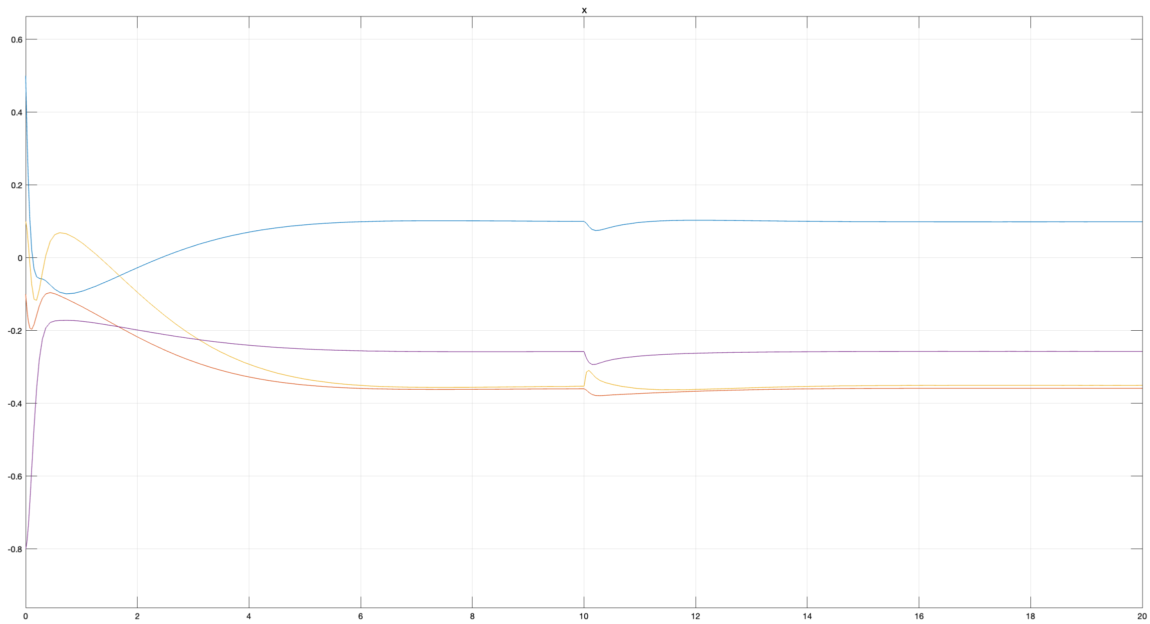


Figure 62 True states

Out put error

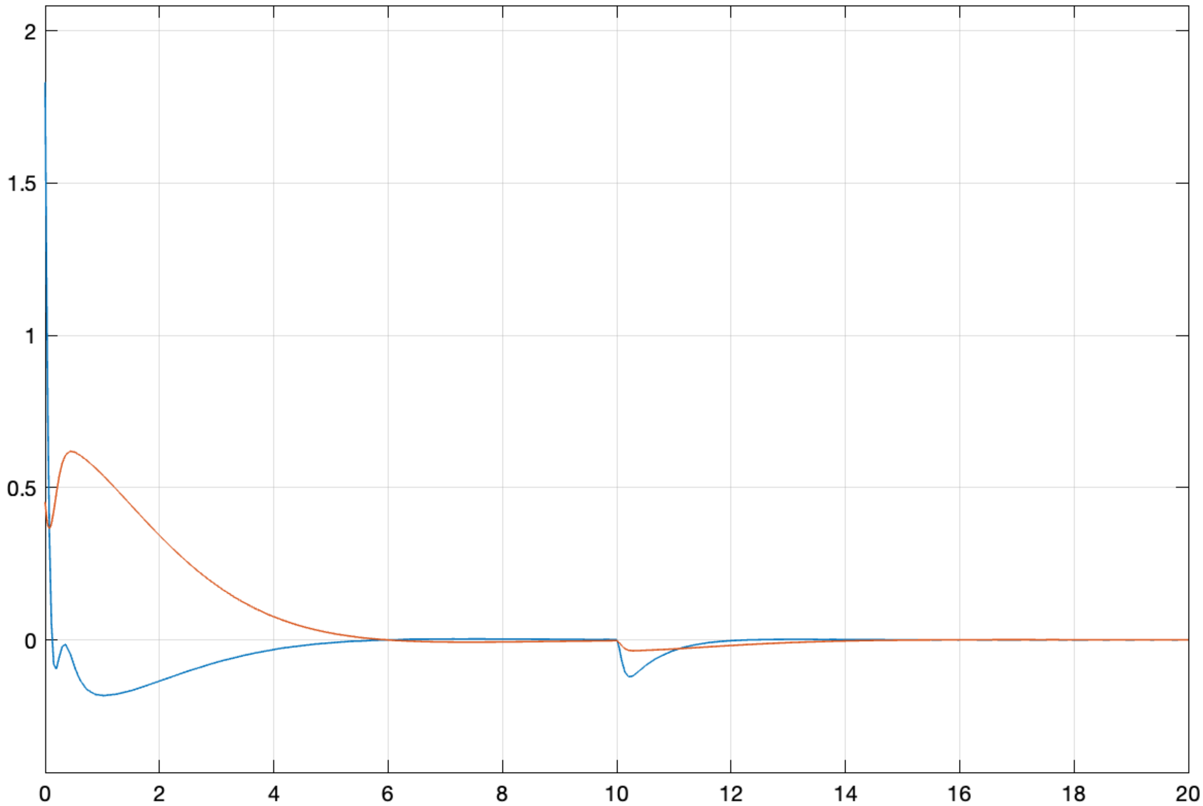


Figure 63 Output error of Servo control

Control signal:

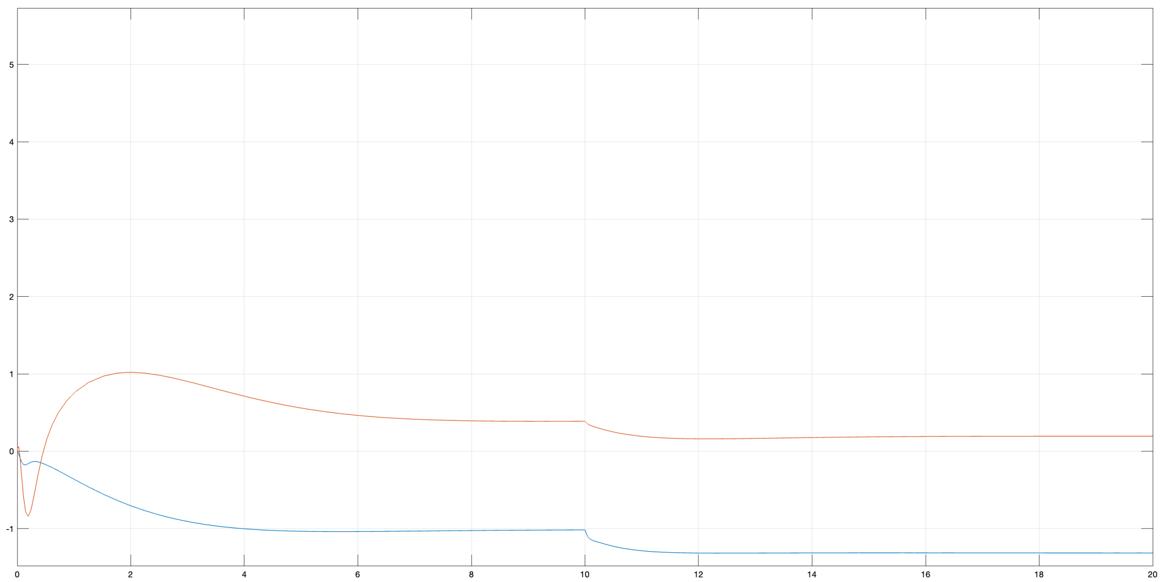


Figure 64 Control signal of Servo control

8.3 Discussion

The simulation results indicate a highly satisfactory performance. The output demonstrates a remarkable accuracy in tracking the target positions. The controller adeptly handles disturbances, swiftly mitigating their impact and enabling the system to return to steady states promptly. Overall, the servo controller exhibits effectiveness in achieving precise and stable control of the system.

9. Exploration

Q6 Can we maintain the states x around a given set points?

I think the answer is **NO** for this system. But it's **YES** for other systems which satisfy the requirements.

My first reaction to this question is do the derivation like LQR. We need do the optimal control, which is similar with LQR approach. Especially for the optimal function

$$J(x_s) = \frac{1}{2}(x_s - x_s p)^T W (x_s - x_{sp})$$

It's in the same form with LQR objective function. Refer to the LQR procedure, I do the derivative as follows:

My student number is A0260014Y, my weight matrix is

$$W = diag(1, 1, 2, 5)$$

The system is:

$$\begin{aligned} u &= -Kx + F \\ \dot{x} &= Ax + B(-Kx + F) \\ &= (A - B)x + Bx_r \end{aligned}$$

$$\begin{aligned} e &= x_r - x \\ x &= x_r - e \end{aligned}$$

Use Lyapunov method:

$$\begin{aligned} \frac{dV(x)}{dt} &= \frac{d(x_s - x_{sp})^T P (x_s - x_{sp})}{dt} \\ &= \dot{x}_s^T P(x_s - x_{sp}) + (x_s - x_{sp})^T P \dot{x}_s \\ &= (Ax + Bu)^T P(x - x_{sp}) + (x + x_{sp}^T P(Ax + Bu)) \\ &= x^T A^T P(x - x_{sp}) + U^T B^T P x_{xp} + x^T PAx - x_{sp} PAx + x^T Bu - x_{sp}^T PAx + x^T Bu - x_{sp}^T Bu + x^T Qx \end{aligned}$$

I lack a clear strategy for solving this function, so I've resorted to utilizing singular values in an attempt to find a solution.

$$\begin{aligned} V(x) &= p(x - x_{sp})^2 \\ \frac{dV(x)}{dt} &= \frac{d(x^2 - 2xx_{sp} + x_{sp})}{dt} \\ &= 2px\dot{x} - 2x_{sp} \\ &= 2px(ax + bu) - 2x_{sp} \\ &= 2pax^2 + 2pxbu - 2x_{sp} \end{aligned}$$

$$u = -kx + x_{sp}$$

$$\frac{dV(x)}{dt} = 2pax^2 + 2pxb(-kx + x_{sp}) - 2x_{sp}$$

I need to find a way to make this equation converge to zero as time approaches infinity. Unfortunately, time constraints are limiting my ability to explore this deeply, and I'll need to allocate more time for a thorough review.

A different perspective

I have an alternative approach, but its efficacy is uncertain. When aiming to individually control states to specific positions, the challenge resembles that of decoupling. In this scenario, if we set the matrix C to be an identity matrix

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

the problem aligns with the decoupling concept, assuming we can accurately measure all parameters. From this standpoint, achieving decoupling and ensuring stability in the separated subsystems could theoretically enable us to control the states individually to the desired positions.

However, despite the ability of this system to achieve decoupling, the outcome is not controllable and unstable, as illustrated in the Chapter 7 Decoupling controller part. **Consequently, for this specific system, achieving the desired goal might be unattainable.**

Indeed, for certain systems that support decoupling control with a stable outcome, this method may work.

10. Discussion and Conclusion

Pole Placement vs. LQR Control:

The pole placement and LQR control methods were both employed to design controllers for the given MIMO system. LQR control demonstrated superior performance in terms of achieving the desired transient response while minimizing control effort. The trade-off between speed and cost was evident, with larger Q values leading to faster convergence but higher control signals.

Observer-Based Control:

The observer-based LQR control approach showcased effective estimation of states, and the selection of observer poles played a crucial role in determining the convergence speed. The trade-off between estimation accuracy and control effort was observed, emphasizing the need for careful pole selection.

Decoupling Controller:

The attempt to design a decoupling controller faced challenges, revealing the system's inherent instability and lack of controllability in the decoupled state. This highlights the importance of thorough analysis and consideration of residual coupling effects between subsystems for successful decoupling.

Servo Control:

The servo controller demonstrated excellent performance in tracking target positions, handling disturbances, and achieving steady states. The simulation results showcased the effectiveness of the control system in maintaining accuracy and stability, essential for practical applications.

Conclusion:

The study and application of various control methods, including pole placement, LQR control, observer-based control, decoupling, and servo control, provided valuable insights into the complex dynamics of the given MIMO system. Each method presented its advantages and challenges, emphasizing the need for a thoughtful approach in selecting control strategies based on the specific requirements and characteristics of the system.

The practical considerations of speed, accuracy, and cost play a crucial role in determining the suitability of control methods for real-world applications. LQR control emerged as a versatile approach, offering a balanced trade-off between achieving rapid convergence, minimizing steady-state errors, and managing control effort.

The challenges faced in decoupling highlight the importance of understanding system dynamics thoroughly and considering practical implications. While theoretical approaches provide valuable insights, real-world applications require a nuanced understanding of system behavior and careful consideration of trade-offs in control strategies.

In conclusion, this mini project has proven highly beneficial in solidifying my understanding of the theories covered in class. It serves as a valuable opportunity for reviewing procedural aspects and preparing for the final exam. I did do all of them step by step following the lecture notes. Most importantly, it has emphasized the practical relevance of the knowledge gained in class, bridging the gap between theoretical concepts and their real-life applications.

Notes: *Due to time constraints, I entered all the equations in Jupyter notebook for ease of typing and then pasted screenshots into this report. Consequently, there might be slight differences in size, and I hope the professor can understand the circumstances.*

Reference

Mainly from **Lecture slides** and **Lecture notes**.

Appendix

(G): [(1.1e+14*(1.5e+26*s^2 + 3.0e+27*s + 8.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(3.2e+18*(3.8e+25*s^2 + 4.5e+26*s + 1.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(2.7e+15*(1.8e+25*s^3 + 3.9e+26*s^2 + 2.5e+27*s - 7.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(1.9e+3*(3.1e+38*s^3 + 7.0e+39*s^2 + 4.9e+40*s + 6.6e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(1.7e+3*(6.2e+37*s^3 + 2.2e+39*s^2 + 2.9e+40*s + 1.3e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(3.3e+29*(2.2e+12*s^2 + 9.7e+12*s - 3.1e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(2.9e+16*(1.8e+25*s^3 + 2.5e+26*s^2 + 9.4e+26*s - 3.7e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(540.0*(1.2e+37*s^2 + 1.6e+40*s + 2.1e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(1.5e+29*(4.4e+12*s^2 + 1.6e+13*s + 1.5e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(5.4e+15*(5.6e+24*s^2 + 1.1e+26*s + 7.9e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(4.6e+3*(3.5e+38*s^2 + 9.9e+39*s + 4.9e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(5.2e+18*(4.4e+25*s^2 + 3.8e+26*s - 5.3e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (7.3e+14*(-
3.3e+25*s^2 + 8.0e+26*s + 5.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (3.2e+13*(1.5e+26*s^2 +
2.6e+27*s + 2.6e+28))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (8.3e+17*(6.3e+25*s^2 + 4.5e+26*s +
3.3e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (2.6e+18*(3.8e+24*s^2 + 2.2e+26*s +
2.1e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44), (6.1e+13*(1.5e+26*s^2 + 3.0e+27*s +
8.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (9.6e+17*(3.8e+25*s^2 + 4.5e+26*s +
1.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (5.2e+14*(1.8e+25*s^3 + 3.9e+26*s^2 + 2.5e+27*s -
7.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (1.1e+3*(3.1e+38*s^3 + 7.0e+39*s^2 + 4.9e+40*s +
6.6e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (2.1e+3*(6.2e+37*s^3 + 2.2e+39*s^2 + 2.9e+40*s +
1.3e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (6.3e+28*(2.2e+12*s^2 + 9.7e+12*s -
3.1e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (8.7e+15*(1.8e+25*s^3 + 2.5e+26*s^2 + 9.4e+26*s -
3.7e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (660.0*(1.2e+37*s^2 + 1.6e+40*s +
2.1e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (4.5e+28*(4.4e+12*s^2 + 1.6e+13*s +
1.5e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (6.5e+15*(5.6e+24*s^2 + 1.1e+26*s +
7.9e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (2.6e+3*(3.5e+38*s^2 + 9.9e+39*s +
4.9e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (1.5e+18*(4.4e+25*s^2 + 3.8e+26*s -
5.3e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (4.2e+14*(- 3.3e+25*s^2 + 8.0e+26*s +
5.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (3.8e+13*(1.5e+26*s^2 + 2.6e+27*s +
2.6e+28))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (1.6e+17*(6.3e+25*s^2 + 4.5e+26*s +
3.3e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (4.9e+17*(3.8e+24*s^2 + 2.2e+26*s +
2.1e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44)]
[(3.4e+18*(3.8e+25*s^2 + 4.5e+26*s + 1.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(1.6e+14*(1.5e+26*s^2 + 3.0e+27*s + 8.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(4.1e+15*(1.8e+25*s^3 + 3.9e+26*s^2 + 2.5e+27*s - 7.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(160.0*(3.1e+38*s^3 + 7.0e+39*s^2 + 4.9e+40*s + 6.6e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(1.8e+3*(6.2e+37*s^3 + 2.2e+39*s^2 + 2.9e+40*s + 1.3e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(9.2e+28*(2.2e+12*s^2 + 9.7e+12*s - 3.1e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(8.2e+15*(1.8e+25*s^3 + 2.5e+26*s^2 + 9.4e+26*s - 3.7e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(48.0*(1.2e+37*s^2 + 1.6e+40*s + 2.1e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(2.3e+29*(4.4e+12*s^2 + 1.6e+13*s + 1.5e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(1.5e+15*(5.6e+24*s^2 + 1.1e+26*s + 7.9e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) +
(4.9e+3*(3.5e+38*s^2 + 9.9e+39*s + 4.9e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) -
(4.6e+17*(4.4e+25*s^2 + 3.8e+26*s - 5.3e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (2.1e+14*(-
3.3e+25*s^2 + 8.0e+26*s + 5.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (4.7e+13*(1.5e+26*s^2 +
2.6e+27*s + 2.6e+28))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (7.3e+16*(6.3e+25*s^2 + 4.5e+26*s +
3.3e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (2.7e+18*(3.8e+24*s^2 + 2.2e+26*s +
2.1e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44), (1.0e+18*(3.8e+25*s^2 + 4.5e+26*s +
1.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (9.1e+13*(1.5e+26*s^2 + 3.0e+27*s +
8.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (7.8e+14*(1.8e+25*s^3 + 3.9e+26*s^2 + 2.5e+27*s -
7.0e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (93.0*(3.1e+38*s^3 + 7.0e+39*s^2 + 4.9e+40*s +
6.6e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (2.2e+3*(6.2e+37*s^3 + 2.2e+39*s^2 + 2.9e+40*s +
1.3e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (1.8e+28*(2.2e+12*s^2 + 9.7e+12*s -
3.1e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (2.4e+15*(1.8e+25*s^3 + 2.5e+26*s^2 + 9.4e+26*s -
3.7e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (58.0*(1.2e+37*s^2 + 1.6e+40*s +
2.1e+41))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (6.7e+28*(4.4e+12*s^2 + 1.6e+13*s +
1.5e+14))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (1.8e+15*(5.6e+24*s^2 + 1.1e+26*s +
7.9e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (2.7e+3*(3.5e+38*s^2 + 9.9e+39*s +
4.9e+40))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (1.4e+17*(4.4e+25*s^2 + 3.8e+26*s -
5.3e+26))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) + (1.2e+14*(- 3.3e+25*s^2 + 8.0e+26*s +
5.8e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (5.7e+13*(1.5e+26*s^2 + 2.6e+27*s +
2.6e+28))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (1.4e+16*(6.3e+25*s^2 + 4.5e+26*s +
3.3e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44) - (5.2e+17*(3.8e+24*s^2 + 2.2e+26*s +
2.1e+27))/(3.1e+42*s^4 + 9.8e+43*s^3 + 1.3e+45*s^2 + 5.5e+45*s + 6.0e+44)]

Related codes:

```
clc
% Q1
% My matriculation number is A0260014Y
clc
clear
close all

a = 0;  b = 0;  c = 1;  d = 4;

A = [ -8.8487+(a-b)/5,         -0.0399,                  -5.55+(c+d)/10,            3.5846;
       -4.574                     2.501*(d+5)/(c+5),   -4.3662,                     -1.1183-(a-c)/20 ;
       3.7698,                    16.1212-c/5,             -18.2103 + (a+d)/(b+4),  4.4936;
     -8.5645-(a-b)/(c+d+2),  8.3742,                    -4.4331 ,                    -7.7181*(c+5)/(b+5) ];

B = [0.0564+b/(10+c),                     0.0319;
       0.0165-(c+d-5)/(1000+20*a),   -0.02;
      4.4939,                                 1.5985*(a+10)/(b+12);
     -1.4269,                                -0.273 ];
C = [-3.2988,           -2.1932+(10*c+d)/(100+5*a),  0.037,      -0.0109;
    0.2922-a*b/500,    -2.1506,                           -0.0104,    0.0163];

% Reference second order system
% Overshoot less than 10%      exp(-pi* sigma)/sqrt(1-sigma^2) < 10%
% Settling time less than 20 seconds    4/(sigma*wn) <20s
syms wn sigma s
equ1 = exp((-pi* sigma)/sqrt(1-sigma^2)) == 10;
res = solve(equ1);
upper_bound =double(res(1));      % 0.5912
%lower_bound =double(res(2));   % -0.5912
%equ2 = sigma <  upper_bound;
%equ3 = sigma >  lower_bound;
sigma = 0.5;
equ4 = 4/(sigma*wn) <20;
%equs = [equ2, equ3, equ4];
res = solve(equ4);  % wn >= 0.4 let wn =0.4

% The reference second order mdoel is 0.16 / (s^2 + 0.4s + 0.16)
equ5 = s^2 + 0.4*s + 0.16 == 0;
poles  = solve(equ5);
% pole 1:  -0.2000 - 0.3464i
% pole 2:  -0.2000 + 0.3464i

%% openloop model
open_loop =  ss(A, B, C, 0);

t=0:0.1:80;
len = size(t,2);
x0_open = [0; 0; 0; 0];
% zero inputs and x0 initial state
x0 = [0.5; -0.1; 0.1; -0.8];
u0=10*zeros(len, 2);
u1=[ones(len,1),zeros(len,1)];
u2=[zeros(len,1),ones(len,1)];

% zero inputs and x0 initial state
[y, tout, x]=lsim(open_loop, u2, t, x0_open);
```

```
figure()
plot(t, y)
grid on
legend('y1','y2')
xlabel('time')
ylabel('output')
title('Open loop step response u2')


step(open_loop)
open_gain = dcgain(open_loop);

%% Use  second-order reference model model
sys1 = ss(A, B, C, 0);
Second_ref_poles = [-2, -2, -0.2000 - 0.3464i,  -0.2000 + 0.3464i]; % set the other two poles 5 times the domonator poles
K_SOD = place(A, B, Second_ref_poles);
Af=A-B*K_SOD;
sys_close=ss(Af, B, C, 0);  %Should be careful about the B here.
step(sys_close)

t=0:0.1:40;
len = size(t,2);

x0 = [0.5; -0.1; 0.1; -0.8];
u0=10*zeros(len,2);
%u1=[ones(len,1),zeros(len,1)];
%u2=[zeros(len,1),ones(len,1)];
u3=[ones(len,1),ones(len,1)];
% zero inputs and x0 initial state
[y, tout, x]=lsim(sys_close, u3, t, x0);

figure()
plot(t,x)
grid on
legend('x1','x2','x3','x4')
xlabel('time')
ylabel('state')
title('zero inputs and x0 initial state')

figure()
plot(t, y)
grid on
legend('y1','y2')
xlabel('time')
ylabel('output')
title('zero inputs and x0 initial state')

% sys_ag_ITAE = ss(A-B*K_SOD,  [0; 0; 0; 1], C, 0);
% DC_gain = dcgain(sys_ag_ITAE);
% dcgain(sys_ag_ITAE);
% F = 1./dcgain(sys_ag_ITAE);
% sys_feedback= F' * sys_ag_ITAE;
% step(sys_feedback)


%% monitor control signal size
```

```
sys1 = ss(A, B, C, 0);
Second_ref_poles = [-2, -2, -0.2000 - 0.3464i,  -0.2000 + 0.3464i]; % set the other two poles 5 times the domonator poles
K_SOD = place(A, B, Second_ref_poles);
Af=A-B*K_SOD;
sys_close_ctr=ss(Af, B, -K_SOD, 0);  %Should be careful about the B here.
step(sys_close_ctr)

t=0:0.1:40;
len = size(t,2);
x0_zeros = [0; 0; 0; 0];
x0_non_zeros = [0.5; -0.1; 0.1; -0.8];
u0=10*zeros(len,2);
%u1=[ones(len,1),zeros(len,1)];
%u2=[zeros(len,1),ones(len,1)];
u3=[ones(len,1),ones(len,1)];
% zero inputs and x0 initial state
[y1, tout1, x1]=lsim(sys_close_ctr, u3, t, x0_zeros);
[y2, tout2, x2]=lsim(sys_close_ctr, u0, t, x0_non_zeros);

figure()
plot(t, y1)
grid on
legend('y1','y2')
xlabel('time')
ylabel('output')
title('Control signal under step reponse situation')

figure()
plot(t, y2)
grid on
legend('y1','y2')
xlabel('time')
ylabel('output')
title('Control signal under non zero state and zeros input')

%% Discuss effects of the positions of the poles on system performance
% set the other two poles 5 times the domonator poles
SOD_poles = [-0.2, -0.2, -0.2000 - 0.3464i,  -0.2000 + 0.3464i;
              -2, -2, -0.2000 - 0.3464i,  -0.2000 + 0.3464i;
             -4, -4, -0.2000 - 0.3464i,  -0.2000 + 0.3464i;
             -10, -10, -0.2000 - 0.3464i,  -0.2000 + 0.3464i];

t=0:0.1:40;
len = size(t,2);
x0 = [0.5; -0.1; 0.1; -0.8];

u0=10*zeros(len,2);
%u1=[ones(len,1),zeros(len,1)];
%u2=[zeros(len,1),ones(len,1)];

%x0 = [0; 0; 0; 0];
%u3=[ones(len,1),ones(len,1)];
% zero inputs and x0 initial state


for i =1:4
```

```
    K_f = place(A, B, SOD_poles(i,:));
    A_f=A-B*K_f;
    sys_close=ss(A_f, B, C, 0);  %Should be careful about the B here.
    [y, ~, x]=lsim(sys_close, u0, t, x0);

    subplot(2,2,i)
    plot(t, y)
    grid on
    legend('y1','y2')
    xlabel('time')
    ylabel('output')
    titles = {['Artifical Poles:' ,num2str(SOD_poles(i,1) )]};
    title(titles)
end

for i =1:4
    K_f = place(A, B, SOD_poles(i,:));
    A_f=A-B*K_f;
    sys_close=ss(A_f, B, C, 0);  %Should be careful about the B here.
    [y, ~, x]=lsim(sys_close, u0, t, x0);

    subplot(2,2,i)
    plot(t, x)
    grid on
    legend('x1','x2','x3','x4')
    xlabel('time')
    ylabel('state')
    titles = {['Artifical Poles:' ,num2str(SOD_poles(i,1) )]};
    title(titles)

end

for i =1:4
    K_f = place(A, B, SOD_poles(i,:));
    A_f=A-B*K_f;
    sys_close=ss(A_f, B, -K_f, 0);  %Should be careful about the B here.
    [y, ~, x]=lsim(sys_close, u1, t, x0);
    subplot(2,2,i)
    plot(t, y)
    grid on
    legend('u1','u2')
    xlabel('time')
    ylabel('state')
    titles = {['Artifical Poles:' ,num2str(SOD_poles(i,1) )]};
    title(titles)
end

    sys_close_ctr=ss(Af, B, -K_SOD, 0);
    [y, tout, x]=lsim(sys_close, u0, t, x0);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  Q2  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Q1=[1 0 0 0
   0 2 0 0
   0 0 3 0
   0 0 0 4];
%
% Q2=[1 0 0 0
```

```
%    0 1 0 0
%    0 0 1 0
%    0 0 0 1 ].*0.5;

R=[1 0
   0 1];

Tau = [A  -B*inv(R)*B';
        -Q1  -A' ];

e = eig(Tau) ;
[V, D] = eig(Tau);
vectors = V(:,[1,2,6,8]);
mu = vectors(5:8, :);
v = vectors(1:4, :);
P = mu*inv(v);
K_LQR = inv(R)*B'*P;

Af=A-B*K_LQR;
sys_close=ss(Af, B, C, 0);  %Should be careful about the B here.
%step(sys_close)

t=0:0.1:40;
len = size(t,2);
x0 = [0; 0; 0; 0];
u0=10*zeros(len,2);
u1=[ones(len,1),zeros(len,1)];
u2=[zeros(len,1),ones(len,1)];
% zero inputs and x0 initial state
[y, tout, x]=lsim(sys_close, u2, t, x0);
figure()
plot(t, y)
grid on
legend('y1','y2')
xlabel('time')
ylabel('output')
title('LQR control step response u2')

%%%%%%%%%%%%%%%%%%%%%%   Q3 Full order %%%%%%%%%%%%%%%%%%%%%%%%
%% evaluate LQR performance when changing Q and R
Q = [0.2 2 4 10];
Q1=[1 0 0 0
   0 2 0 0
   0 0 3 0
   0 0 0 4];
%R=[1 0
   %0 1].*0.5;

R=[1 0
   0 1];
K_LQR = lqr(A, B, Q1, R);
% use LQR to design estimator
[L_LQR,~,~] = lqr(A', C', Q1, R);
L_LQR = L_LQR';

% use poles placement to design estimator
desired_poles = [-4, -5, -6, -7];
```

```
L_2 = place(A', C', desired_poles);
L_estimator = L_2';

sys_ag_LQR = ss(A-B * K_LQR, B , C, 0);

t=0:0.1:40;
len = size(t,2);
x0 = [0.5; -0.1; 0.1; -0.8];
u0=10*zeros(len,2);
%u1=[ones(len,1),zeros(len,1)];
%u2=[zeros(len,1),ones(len,1)];
u3=[ones(len,1),ones(len,1)];
% zero inputs and x0 initial state
[y, tout, x]=lsim(sys_ag_LQR, u3, t, x0);

figure()
plot(t,x)
grid on
legend('x1','x2','x3','x4')
xlabel('time')
ylabel('state')
title('zero inputs and x0 initial state')

figure()
plot(t, y)
grid on
legend('y1','y2')
xlabel('time')
ylabel('output')
title('zero inputs and x0 initial state')


%%%%%%%%%%%%%%%%%%%%%%%%   Q3 Reduced order %%%%%%%%%%%%%%%%%%%%%%%%%%
Wo = [C; C*A; C*A^2; C*A^3];
syms l11 l12 l13 l14 l21 l22 l23 l24 s
L = [l11 l12 l13 l14; l21 l22 l23 l24]';
I = eye(4);
close_sys = s*I - (A-L*C);
det(close_sys)

%% reduced order observer
syms t1 t2 t3 t4 t5 t6 t7 t8
T = [t1 t2 t3 t4; t5 t6 t7 t8];
CT = [C; T];



%%%%%%%%%%%%%%%%%%%%%%%%   Q5 Servo control + LQR  %%%%%%%%%%%%%%%%%%%%%%
w = [0.3; 0.2];

% verify controlbility
Qc = [A B;C zeros(2,2)];
assert(rank(Qc)==6);

A_bar=[A zeros(4,2);
       -C zeros(2,2)];
```

```
B_bar=[B; zeros(2,2)];
B_w_bar=[B; zeros(4,2)];
B_r_bar=[zeros(4,2); eye(2)];
C_bar=[C, zeros(2,2)];

Q=[1 0 0 0 0 0;
      0 1 0 0 0 0;
     0 0 1 0 0 0;
    0 0 0 1 0 0;
   0 0 0 0 1 0;
  0 0 0 0 0 1]*10;

R=[1 0
     0 1]*1;

gamma=[A_bar -B_bar/R*(B_bar');-Q -A_bar'];

[eig_vector,eig_value]=eig(gamma);
eig_value_sum=sum(eig_value);
vueogen=eig_vector(:,real(eig_value_sum)<0);
P=vueogen(7:12,:)/vueogen(1:6,:);
K_calculated=real(inv(R)*(B_bar')*P);

K1=K_calculated(:,1:4);
K2=K_calculated(:,5:6);

%% full order Observer LQR method
% A_ba = A', B_ba = C', K_ba = L'
Qbar=[1 0 0 0
      0 1 0 0
      0 0 1 0
      0 0 0 1]*5;
Rbar=[1 0
       0 1 ]*1;
x0 = [0.5; -0.1; 0.1; -0.8];
Phi1 = [A',-C'/Rbar*C;-Qbar,-A];
[eig_vector_observed,eig_value_observed]=eig(Phi1);
eig_value_observed_sum=sum(eig_value_observed);
vueigen_observered=eig_vector_observed(:,real(eig_value_observed_sum)<0);
P_observed=vueigen_observered(5:8,:)/vueigen_observered(1:4,:);
Kbar=real(inv(Rbar)*C*P_observed);
L=Kbar';
L=real(L);
```